A Simple Coded ARQ for Satellite Broadcasting

Gianluigi Liva, Christian Kissling, and Christoph Hausl

Abstract: **We introduce a novel packet retransmission technique which improves the efficiency of automatic retransmission query (ARQ) protocols in the context of satellite broadcast/multicast systems. The proposed coded ARQ technique, similarly to fountain coding, performs transmission of redundant packets, which are made by linear combinations of the packets composing the source block. Differently from fountain codes, the packets for the linear combinations are selected on the basis of the retransmission requests coming from the user terminals. The selection is performed in a way that, at the terminals, the source packets can be recovered iteratively by means of simple back-substitutions. This work aims at providing a simple and efficient alternative to reliable multicast protocols based on erasure correction coding techniques.**

Index Terms: **Automatic retransmission query (ARQ), broadcast/multicast networks, coding, fountain codes, network coding.**

I. INTRODUCTION

Automatic retransmission query (ARQ) protocols [1] are indeed a very simple countermeasure against packet losses in wireless communication systems. However, the adoption of ARQ techniques in a broadcast/multicast system usually leads to a degradation of the performance, especially in case of uncorrelated packet losses at the receivers. Consider the example depicted in Fig. 1. Here, the broadcasting system is composed by a satellite, a gateway, and a group of four users. The four users lost four different packets, hence four retransmissions are needed. For a system with many users, this approach may hence lead to a large number of retransmissions.

Efficient alternatives to ARQ protocols for broadcast/multicast, based on erasure recovery, were proposed in [2]–[4]. Several other algorithms inspired by the concept of network coding [5] were considered in [6]–[8]. Further coded approaches have been proposed, which are based on either the so-called fountain codes [9]–[13] or on sparse-graph block codes for erasure correction [14], [15]. Here, redundant packets are produced by the transmitter for a set of k source packets. The redundant packets (which are linear combinations of source packets) are exploited by the receiver to recover the lost ones. If a receiver succeeds in recovering the k source packets, it signals the decoding success to the transmitter. Once all the receivers accomplished this task, the encoder stops producing redundancy. The encoding procedure then restarts for the next set of k source packets. Although originally proposed in the context of iterative decoding [9], [12], the fountain coding approach becomes very efficient when maximum-likelihood (ML) decoders are used to recover the source block. Usually, with ML decoding a few packets more than k have to be collected by each receiver to recover the source block. Note that for fountain codes based on sparse matrices ML decoding can be achieved by means of moderate-complexity algorithms performing the inversion of a binary sparse matrix [11], [15]–[17]. However, for terminals limited in computational power, this approach may indeed be still impractical.

The solution proposed in this paper, which will be referred to as coded ARQ (C-ARQ) [18], finds a low-complexity compromise between the above-presented solutions. As for selective ARQ (S-ARQ), just packets signalled as missing at the users side are retransmitted; as for fountain/block codes, the redundant packets are not simply copies of original ones, but they are a linear combination of many lost packets. Contrary to algorithms presented previously [6]–[8], the proposed algorithm uses a combining window to restrict the latency and it takes into account the number of users missing a packet and gives priority to retransmissions of packets that are missed by many users. Moreover, it excludes packets from retransmissions with the help of a taboo marker in order to allow the users to decode with low computational complexity.

The paper is organized as follows. In Section II the system model is described. Section III details the proposed coded ARQ approach. A performance evaluation is provided in Section IV. Conclusions follow in Section V.

II. SYSTEM OVERVIEW

We focus on the architecture depicted in Fig. 1. We consider a satellite gateway, a set of N users, and a satellite connecting users and gateway. The satellite broadcasts the packets received from the gateway. The packets are here assumed of constant size. Moreover, the packets are labeled in a way they can be unambiguously identified (e.g., by mean of a packet counter). A reliable feedback channel, used to signal the retransmission requests, is supposed to be available. Note that the feedback channel does not need to be satellite-based (i.e., a terrestrial wireless network could be used to carry the terminal feedbacks). The packets can be of two types: Source or redundancy packets. Redundancy packets are produced on the basis of the retransmission requests received by the gateway. They are built by bit-wise sums of source packets (or eventually of the preceding redundancy packets). The labels of the packets involved in the construction of a redundancy packet are signalled to the users.

III. CODED ARQ RETRANSMISSION

The proposed C-ARQ approach deals with the construction of redundancy packets by the bit-wise sum of the source packets for which a retransmission has been requested. The relation between the source packets involved in the linear combination

Manuscript received April 14, 2010.

G. Liva and C. Kissling are with the Institute of Communication and Navigation of the Deutsches Zentrum für Luftund Raumfahrt (DLR), 82234 Wessling, Germany, email: {Gianluigi.Liva, Christian.Kissling}@dlr.de.

C. Hausl is with the Institute for Communications Engineering, Technische Universität München, 80290 Munich, Germany, email: Christoph.Hausl@tum.de.

and the resulting redundant packet can be regarded as a parity equation. Consider the case where a generic user requested the retransmission of a packet involved in a parity equation. He would be able to recover the missing packet by performing a bit-wise sum of the associated redundant packet and the other (complementary) source packets involved in the same equation. The above-described procedure will succeed if and only if the user possesses all the complementary packets together with the redundant one. The latency introduced by this approach is given by the maximum distance (in time) between a packet involved in an equation and the related redundancy packet. To keep a control on the latency, we introduce therefore the concept of combining window. A combining window represents the set of L packets over which a parity-check equation can be built. The time axis is therefore partitioned in blocks of L packets (see Fig. 2). Alternatively, one could select the packets involved in the equation within the L-packets window beginning with the 1st missing packet of the equation.

Let us denote by $\{\rho_k\}$, $k = 1,\cdots,L$, the set of L source packets within a combining window. We define by Ψ_i the set of source packets lost by the ith user within the window. The number of packets lost by the ith user is $q_i = |\Psi_i|$. We denote by $\{\phi_k\}$, $k = 1,\cdots,R$, the set of redundant packets built to recover the source packets lost in the combining window. Each redundant packet is obtained as (bit-wise) linear combination of the source packets, i.e.,

$$\phi_k = a_{k,1}\rho_1 \oplus a_{k,2}\rho_2 \oplus a_{k,3}\rho_3 \oplus \cdots \oplus a_{k,L}\rho_L = \bigoplus_{j=1}^{L} a_{k,j}\rho_j$$

with $a_{k,j} \in \{0,1\}$. R gives a measure of the retransmissions overhead, since it represents the number of redundant packets produced for a combining window. With fountain codes, linear combinations of source packets are performed randomly, following certain probability distributions. The main difference with the approach proposed in this paper resides in the fact that we exploit the knowledge of the erasure pattern of each user to build ad-hoc parity equations.

The ith user needs to solve the system of binary equations $\mathbf{A}\rho = \phi$, i.e.,

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,L} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,L} \\ \vdots & \vdots & \ddots & \vdots \\ a_{R,1} & a_{R,2} & \cdots & a_{R,L} \end{pmatrix} \begin{pmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \vdots \\ \rho_L \end{pmatrix} = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_R \end{pmatrix} \tag{1}$$

with unknowns in Ψ_i. This can be achieved if and only if $\text{rank}\mathbf{A} = \mathbf{q_i}$. Hence, R can be lower bounded as

$$R \geq \max_i\{q_i\}, \tag{2}$$

the number of retransmissions cannot be lower than the maximum amount of packets lost by a user.[1]

[1]Note that the bound of (2) holds also for fountain codes: The user experiencing the maximum number of erasures ($q_M \doteq \max_i\{q_i\}$) needs at least q_M redundant packets to recover the unknowns (this follows from the fact that a consistent system of equations with less equations than unknowns admits multiple solutions).

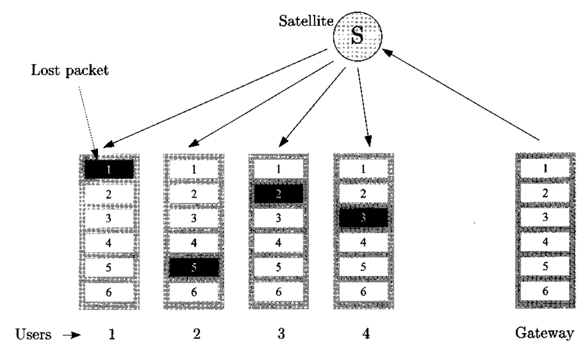


Fig. 1. Overview of the system architecture.

A. Retransmission Patterns Selection - Algorithm A (Taboo Algorithm)

We introduce next a retransmission pattern selection algorithm which takes care that the equations system (1) can be solved by each user by simple back-substitutions. Here, resides a crucial difference with fountain/block codes, where the equations system is usually solved by smart Gaussian elimination algorithms [11], [15]–[17], with indeed larger complexity w.r.t. the approach proposed herein. In fact, for random Luby-transform (LT) [10] and random linear block codes operating of blocks of k source symbols, the decoding complexity (via Gaussian elimination) scales as $\mathcal{O}(k^3)$. Practical maximum-likelihood implementations of Raptor [10], [11] and low-density parity-check (LDPC) code decoders [15]–[17] usually exploit efficient (smart) Gaussian elimination algorithms to reduce the decoding complexity. The reduction in complexity is due to the sparseness of the equation system imposed on the codeword symbols. However, in the algorithms of [11], [16] the final step still turns into the solution of a dense system of equations involving a reduced number of unknowns, e, with e being a fraction of the source block size k (for Raptor codes) or a fraction of the codeword length n (for LDPC codes). The complexity of this decoding stage scales as $\mathcal{O}(e^3)$. The solution of a system of equations (in triangular form) in k unknowns via back-substitutions has complexity scaling as $\mathcal{O}(k^2)$. The possibility of solving the equations system by back-substitutions is ensured by excluding packets from the retransmission patterns with the help of a taboo marker.

Initialization phase. An error count vector is initialized, $\mathbf{e} = \{\mathbf{e_1}, \mathbf{e_2}, \cdots, \mathbf{e_L}\}$ where e_i is the number of users missing the ith packet ($0 \leq e_i \leq N$). An auxiliary vector $\mathbf{t} = \{\mathbf{t_1}, \mathbf{t_2}, \cdots, \mathbf{t_L}\}$ is created. Each element of $\mathbf{t}$ takes a value within the set of flags (T, M, K). An element t_i is set to K (*known*) if ρ_i has been received by all the users (i.e., if $e_i = 0$), otherwise it is set to M (*missing*). The redundant packet index k is initialized to 1.

Construction of redundant packets. Repeat (1-5) until $t_i = K, \forall i = 1,\cdots,L$.

1. For $j = 1,\cdots,L$, set $a_{k,j} = 0$.

2. Repeat (a-c) until $t_i \neq M, \forall i = 1,\cdots,L$

   a. Find $l = \arg\max_i\{e_i\}$ under the constraint $t_i = M$.

   b. Set $a_{k,l} = 1$, $e_l = 0$, and $t_l = K$.

   c. For $j = 1,\cdots,N$, if $\rho_l \in \Psi_j$, for all i s.t. $\rho_i \in \Psi_j$ and $t_i = M$, set $t_i = T$ (*taboo*).

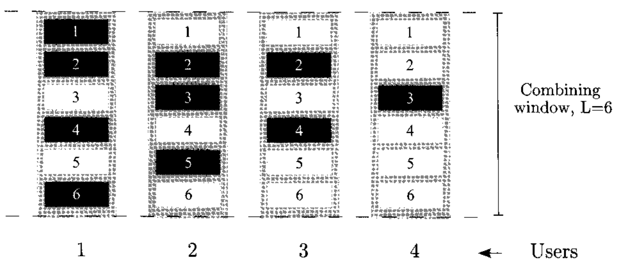


Fig. 2. Example of lost packet patterns for the case of $N = 4$ users. Combining window length $L = 6$.

3. Obtain the kth redundant packet as $\phi_k = \bigoplus_{j=1}^{L} a_{k,j}\rho_j$.
4. For all $\forall i = 1, \cdots, L$, if $t_i = T$, set $t_i = M$.
5. Increment k by 1.

Note that the algorithm stops whenever all the elements of $\mathbf{t}$ are marked as K, i.e., if all the lost packets can be recovered from the set of redundant packets.

Example. Consider again the case of Fig. 2. Initially, the vector $\mathbf{e}$ is set as $\{1,3,2,2,1,1\}$, while $\mathbf{t}=\{\mathbf{M},\mathbf{M},\mathbf{M},\mathbf{M},\mathbf{M},\mathbf{M}\}$. We begin with the construction of the first redundant packet (k is initialized to 1). The second packet is the one with the highest value in $\mathbf{e}$ (check that $\arg\max_i\{e_i\} = 2$). The vector $\mathbf{e}$ is modified as $\{1,0,2,2,1,1\}$. The second packet was lost by the users $1, 2$, and 3. The other packets lost by these users have to be marked as taboo. Hence, $\mathbf{t} = \{\mathbf{T},\mathbf{K},\mathbf{T},\mathbf{T},\mathbf{T},\mathbf{T}\}$. Thus, $\phi_1 = \rho_2$. For the second redundant packet ($k = 2$), we have to reset $\mathbf{t} = \{\mathbf{M},\mathbf{K},\mathbf{M},\mathbf{M},\mathbf{M},\mathbf{M}\}$. Now, there are two packets with the highest value in $\mathbf{e}$: ρ_3 and ρ_4. Assuming we select the first one, $\mathbf{e}$ is modified as $\{1,0,0,2,1,1\}$ and $\mathbf{t} = \{\mathbf{M},\mathbf{K},\mathbf{K},\mathbf{M},\mathbf{T},\mathbf{M}\}$. In the next iteration, ρ_4 is selected ($t_4 = M$ and $\arg\max_i\{e_i\} = 4$). Hence, $\mathbf{e}$ becomes $\{1,0,0,0,1,1\}$ and $\mathbf{t} = \{\mathbf{T},\mathbf{K},\mathbf{K},\mathbf{K},\mathbf{T},\mathbf{T}\}$. The second redundant packet is thus $\phi_2 = \rho_3 \oplus \rho_4$. Going on with the procedure, the other redundant packets will be $\phi_3 = \rho_1 \oplus \rho_5$ and $\phi_4 = \rho_6$. Remarkably, for this example, the proposed algorithm performs optimally, i.e., it achieves the minimum number of retransmissions ($R = 4$) according to (2).

IV. PERFORMANCE EVALUATION

We provide next a performance evaluation for the proposed C-ARQ scheme. We first derive a lower bound to the average number of retransmissions based on (2). Let's denote by Q_i the random variable (r.v.) related to the number of lost packets by the ith user within a window. We define the r.v. related number of required retransmissions by $\mathcal{R} = \max\{Q_i\}$ (according to (2)). The average number of retransmissions for an ideal scheme which satisfies (2) with equality is

$$\mathbb{E}[\mathcal{R}] = \sum_{R=1}^{L} R \Pr\{\mathcal{R} = R\} \quad (3)$$

where $\Pr\{\mathcal{R} = R\}$ is the probability that the worst user experiences R packet losses. We remark that (3) represents a lower bound to the average number of retransmissions required by any retransmission technique. We focus now on the case where the packet losses are uncorrelated in time and among users. We further assume that the generic ith user experiences packet loss probability ϵ_i. The term $\Pr\{\mathcal{R} = R\}$ can be expressed as

$$\Pr\{\mathcal{R} = R\} = \Pr\{Q_i < R+1, \forall i\} - \Pr\{Q_i < R, \forall i\}$$

which, considering the users independency, reduces to

$$\Pr\{\mathcal{R} = R\} = \prod_{i=1}^{N} \Pr\{Q_i < R+1\} - \prod_{i=1}^{N} \Pr\{Q_i < R\} \quad (4)$$

where $\Pr\{Q_i < R\}$ is given by

$$\Pr\{Q_i < R\} = \sum_{w=0}^{R-1} \binom{L}{w} \epsilon_i^w (1-\epsilon_i)^{L-w}. \quad (5)$$

Consider the case of S-ARQ. The probability that a packet has to be retransmitted is given by $P_{\text{S-ARQ}} = 1 - \prod_{i=1}^{N}(1-\epsilon_i)$. The average number of retransmissions in a window is

$$\mathbb{E}[\mathcal{R}_{\text{S-ARQ}}] = LP_{\text{S-ARQ}} = L\left[1 - \prod_{i=1}^{N}(1-\epsilon_i)\right]. \quad (6)$$

For the special case where the users experience the same loss probability, $\epsilon_i = \epsilon, \forall i = 1, \cdots, N$, (4), (5), and (6) simplify to

$$\Pr\{\mathcal{R} = R\} = [\Pr\{Q_i < R+1\}]^N - [\Pr\{Q_i < R\}]^N, \quad (7)$$

$$\Pr\{Q_i < R\} = \sum_{w=0}^{R-1} \binom{L}{w} \epsilon^w (1-\epsilon)^{L-w}, \quad (8)$$

$$\mathbb{E}[\mathcal{R}_{\text{S-ARQ}}] = LP_{\text{S-ARQ}} = L[1 - (1-\epsilon)^N]. \quad (9)$$

In Fig. 3, the average number of retransmissions per combining window (normalized to L) is depicted as a function of the window size, L. The chart reports the performance according to (3) and (6), considering $N = 100$ users and a packet loss probability $\epsilon = 10^{-2}$. Simulation results for algorithm A are provided as well. A large reduction of the number of retransmissions is observed for the C-ARQ. For $L = 800$, nearly 97% of the retransmission required by S-ARQ could be spared. Note that the algorithm A tightly approaches the bound of (3). The performance of C-ARQ tends to saturate for large values of L, making it reasonable to keep L in the range of $100 - 200$ for limiting the system latency.

In Fig. 4, the average number of retransmissions per window (normalized to L) is depicted as a function of the packet loss probability ϵ, for the case of $L = 100$ and $N = 100$. The chart reports the performance according to (3) and (6) and simulation results for algorithm A. Again, the proposed algorithm almost matches the bound of (3). The advantage of the proposed technique is rather evident for moderate-to-high packet loss rates (i.e., $\epsilon > 10^{-3}$), while for low packet loss rates the performances of C-ARQ and of S-ARQ tend to converge. (For very small ϵ, most of the times only one packet is lost within a combining window, and both C-ARQ and S-ARQ would both perform a single retransmission).

Fig. 5 depicts the average number of retransmissions per combining window (normalized to L) as a function of the window

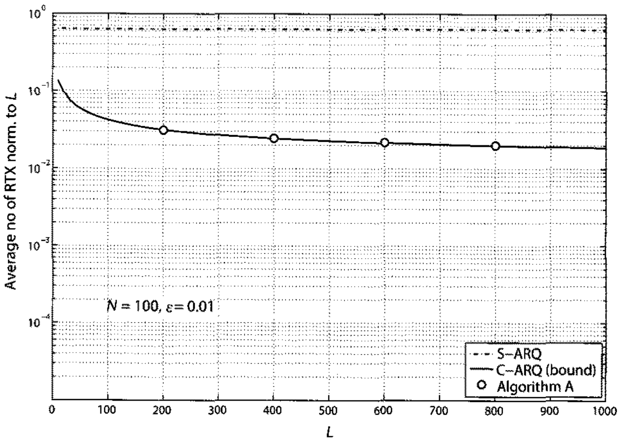


Fig. 3. Normalized average number of retransmissions per combining window as a function of the window size L. $\epsilon = 10^{-2}$, $N = 100$ users.

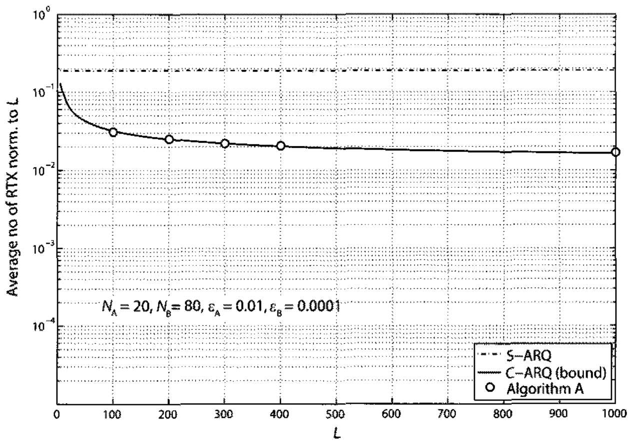


Fig. 5. Normalized average number of retransmissions per combining window as a function of the window size L. Population of $N = 100$ users divided in two subsets. $N_A = 20$ users with loss probability $\epsilon_A = 10^{-2}$, $N_B = 80$ users with loss probability $\epsilon_B = 10^{-4}$.

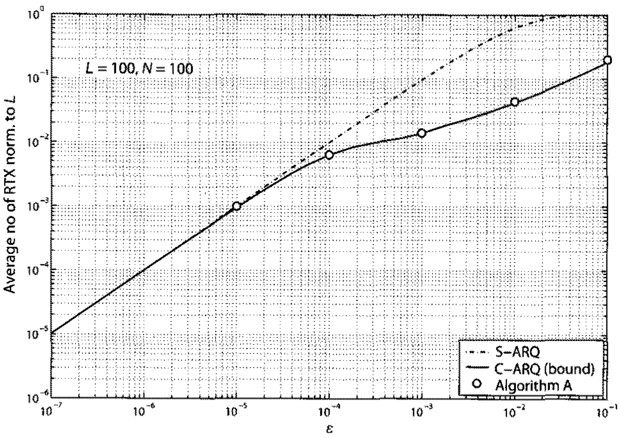


Fig. 4. Normalized average number of retransmissions per combining window as a function of ϵ. $L = 100$, $N = 100$ users.

size, L. The performance has been obtained according to (3) and (6), considering a population $N = 100$ users divided in two subsets of $N_A = 20$ and $N_B = 80$ users, respectively, where the first set of users is characterized by a packet loss probability $\epsilon_A = 10^{-2}$. The second set of users experiences a much lower loss probability $\epsilon_B = 10^{-4}$. The performance is here dominated by the average number of retransmissions per combining window required by the users experiencing the worse channel conditions (note in fact that for S-ARQ $\mathbb{E}\left[\mathcal{R}_{\text{S-ARQ}}\right]/L = 0.1886$ for $L \geq 1$, and reduces to $\mathbb{E}\left[\mathcal{R}_{\text{S-ARQ}}\right]/L = 0.1821$ when only the first subset of $N_A = 20$ users is considered). Also in this case, the algorithm A tightly approaches the bound of (3).

V. CONCLUDING REMARKS

In this paper we introduced a novel efficient C-ARQ approach for satellite broadcast/multicast systems, which permits limiting the retransmissions by broadcasting linear combinations of the requested packets. A lower bound on the average number of retransmissions is introduced. A simple algorithm for deriving the linear combinations is provided, which almost achieves this lower bound. The algorithm allows the users to recover the source packets iteratively by means of simple back-substitutions, hence with much lower complexity w.r.t. fountain schemes based on ML decoders. Simulation results confirm that the saving in terms of retransmission number is large w.r.t. S-ARQ, especially in case of high packet loss rate regimes.

REFERENCES

[1] D. Bertsekas and R. Gallager, *Data networks*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1987.

[2] J. Metzner, "An improved broadcast retransmission protocol," *IEEE Trans. Commun.*, vol. 32, pp. 679–683, June 1984.

[3] J. Nonnenmacher, E. W. Biersack, and D. Towsley, "Parity-based loss recovery for reliable multicast transmission," *IEEE/ACM Trans. Netw.*, vol. 6, No. 4, pp. 349–361, Aug. 1998.

[4] B. Adamson, C. Bormann, M. Handley, and J. Macker, "NACK-oriented reliable multicast protocol," IETF (RFC 3940).

[5] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 4, no. 4, pp. 1204–1216, July 2000.

[6] L. Keller, E. Drinea, and C. Fragouli, "Online broadcasting with network coding," in *Proc. IEEE NETCOD*, Hong Kong, Jan. 2008.

[7] J. K. Sundararajan, D. Shah, and M. Médard, "ARQ for network coding," in *Proc. IEEE ISIT*, Toronto, July 2008.

[8] R. A. Costa, D. Munaretto, J. Widmer, and J. Barros, "Informed network coding for minimum decoding delay," in *Proc. IEEE MASS*, Atlanta, Sept. 2008.

[9] J. Byers, M. Luby, and M. Mitzenmacher, "A digital fountain approach to reliable distribution of bulk data," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 8, pp. 1528–1540, Oct. 2002.

[10] M. Shokrollahi, "Raptor codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2551–2567, June 2006.

[11] 3GPP TS 26.346 V6.1.0, *Technical specification group services and system aspects; multimedia broadcast/multicast service; protocols and codecs*, June 2005.

[12] M. Luby, "LT codes," in *Proc. Symp. Foundations of Comput. Science*, Washington, DC, USA, 2002.

[13] G. Liva, E. Paolini, and M. Chiani, "Performance versus overhead for fountain codes over $\mathbb{F}_q$," *IEEE Commun. Lett.*, vol. 14, no. 2, pp. 178–180, 2010.

[14] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. Spielman. "Efficient erasure correction codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, Feb. 2001.

[15] E. Paolini, G. Liva, M. Varrella, B. Matuz, and M. Chiani, "Low-complexity LDPC codes with near-optimum performance over the BEC," in *Proc. ASMS*, Bologna, Aug. 2008.

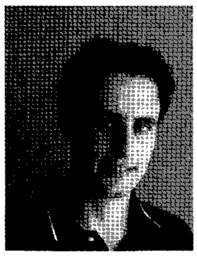
[16] D. Burshtein and G. Miller, "An efficient maximum likelihood decoding of LDPC codes over the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, Nov. 2004.

[17] B. Matuz, G. Liva, E. Paolini, and M. Chiani, "Pivoting algorithms for

maximum likelihood decoding of LDPC codes over erasure channels," *IEEE GLOBECOM*, Hawaii, Nov. 2009.
[18] G. Liva and C. Kissling, "Verfahren zur broadcasting-übertragung von in datenpaketen angeordneten informationen," Deutsches Patent- und Markenamt, Patent application 10 2008 003 588.2, Jan 2008.

Gianluigi Liva was born in Spilimbergo, Italy, on July 23rd, 1977. He received the Laurea in Electronic Engineering in 2002, and the Ph.D. degree in 2006 at DEIS, University of Bologna (Italy). His main research interests include satellite communication systems, random access techniques, and error control coding. Since 2003, he has been involved in the research of channel codes for high data rate Consultative Committee for Space Data Systems (CCSDS) missions, in collaboration with the European Space Operations Centre of the European Space Agency (ESA-ESOC). From October 2004 to April 2005, he was researching at the University of Arizona as visiting student, where he was involved in the design of low-complexity coding systems for space communication systems. He is currently with the Institute of Communications and Navigation at the German Aerospace Center (DLR). He is active in the DVB-SH and in the DVB-RCS Mobile standardization groups. In 2010, he has been appointed guest lecturer for channel coding at the Institute for Communications Engineering (LNT) of the Technische Universität München (TUM). He is IEEE member and he serves IEEE as reviewer for Transactions, Journals, and Conferences.

Christian Kissling studied at the Technische Universität München (TUM), Germany and the Carnegie Mellon University Pittsburgh (CMU), USA and received his Dipl.-Ing. degree from the TUM in Electrical Engineering, specializing in Information and Communication Technology in 2005. Since 2005, he is with the Institute of Communications and Navigation of the German Aerospace Center (DLR) where he works as scientific researcher in aeronautics as well as space communication projects. He is author of numerous publications and has been involved in many projects of the European Space Agency (ESA) and the European Commission (EC) as well as standardization bodies including ICAO. His current research interests are in the field of radio resource management for satellite systems, in particular random access and DAMA techniques as well as QoS management, networking, and aeronautical communication.



Christoph Hausl received the Dipl.-Ing. and Dr.-Ing. degree in Electrical Engineering and Information Technology from Technische Universität München, Germany in 2004 and 2008, respectively. Since 2004, he has been with the Institute for Communications Engineering at the Technische Universität München as a research and teaching assistant. He is co-recipient of Best Paper Awards at the International Conference on Communications 2006 and at the International Workshop on Wireless Ad-hoc and Sensor Networks 2006. He served as guest editor for the special issue on Physical Layer Network Coding for Wireless Cooperative Networks of the EURASIP Journal on Wireless Communications and Networking in 2010. His current research interest include channel coding and network coding, and their application to wireless relay networks and mobile communications.