A New Coding Scheme for the Noisy-Channel Slepian-Wolf Problem: Separate Design and Joint Decoding

Ruiyuan Hu, Ramesh Viswanathan and Jing (Tiffany) Li
Electrical and Computer Engineering Dept., Lehigh University, Bethlehem, PA 18105
Email: {ruh2, rav7, jingli}@ece.lehigh.edu

***Abstract*— In this paper, a new scheme for solving the Slepian-Wolf problem over noisy channels using serially concatenated codes is proposed. An outer low density parity check code is used to perform distributed source coding, and an inner convolutional code adds error protecting capability to the compressed data. Soft iterative joint source-channel decoder is performed, where the decoder side information is provided to the sub outer decoder instead of the sub inner decoder. The scheme is attractive since separate refining of compression rate (outer code) and error protection power (inner code) makes the design easy and the performance controllable, and joint iterative decoding exploits the power of serial concatenated structure as much as possible. Simulations reveal encouraging joint decoding gain especially at low signal-to-noise ratios.**

I. Introduction

The Slepian-Wolf theorem [1], which forms the basis of lossless distributed source coding (DSC) problem, defines the achievable rate region when two physically separated and statistically correlated sources are independently encoded and jointly decoded for a lossless channel. The binning/coset/syndrome approach used in the proof of the Slepian-Wolf theorem provides a fresh and sharp tool for powerful linear channels codes to be optimally exploited in distributed source coding [1][2]. The first constructive realization of the Slepian-Wolf boundary using practical channel codes was proposed in [3] where single-parity check codes were used with the binning scheme. Advanced formulations using powerful turbo codes (e.g. [5]-[9]) and low density parity check (LDPC) codes (e.g. [13][14]) were subsequently proposed to solve the lossless-channel DSC problem.

When the transmission channel is noisy (as in a practical case), the problem becomes more involved since it now requires an *error-resilient distributed source coding* solution. Excellent schemes have been proposed employing turbo codes [10]-[12] and LDPC/LDPC-like codes [15] for noisy-channel DSC. The idea, first demonstrated in [10], is to absorb compression in channel coding and to use a single channel code for the dual purpose of source and channel coding. The efficiency and efficacy of this idea is best exemplified in [15], where density evolution is exploited to design and optimize irregular repeat accumulate (IRA) codes to match to the channel. While the performance in [15] is impressive, it should be noted that density evolution targets asymptotic performance (or very large block sizes), warranting little optimality for short block sizes like a few hundred bits to a few thousand bits (which are typical in practical applications).

In this paper, we propose to attack the noisy-channel DSC problem using serially concatenated codes (SCC). The source-channel separation theorem states that separate DSC and channel coding need not incur loss in capacity compared to joint source-channel coding, provided both parts are done optimally [16]. While this theorem sets us free from the very tricky and tedious task of joint design, due to various limitations and constraints, however, sub-optimal codes are often used, causing a performance degradation. For example, in practical systems using separate source and channel coding, the residual errors at the output of the channel decoder can often be catastrophic to the source decoder. It is this dilemma of for-and-against that inspires the idea of *separately designing a good source code and a good channel code, and unifying them using one combined decoder*. Serially concatenated codes, which have two separate sub-codes that are jointly decoded using *one* soft iterative decoder, and which are among the most powerful coding schemes known today, fit naturally into the scenario. Unlike the single-code approach [10]-[12] which provides little information on how code rate is allocated between compression and error protection, the proposed approach provides the freedom to independently selecting compression and error protection rates, making the performance predictable and controllable. At the same time, the use of joint iterative decoding makes it possible to harness the power of serially concatenated structure as much as possible.

Realizing the power of the proposed idea requires that both sub codes be soft decodable. This can be tricky for the outer code, since it is a channel code that assumes the role of source coding. If the binning framework is used (the only generic and provenly optimal framework for DSC, details follow in Section II), then this outer code will map data sequences (viewed as virtual codewords) to their respective syndromes to perform compression. Hence, to enable effective soft decoding at the decoder, it is important that the reliability information of the syndrome sequences can be exploited in decoding. This is difficult for a general linear channel code including turbo codes. Fortunately, the unique features in the structure and the decoding method of LDPC codes makes it not only feasible, but also efficient. As will be discussed later, by exploiting an *extended* parity check matrix, the same message-passing algorithm can be used to soft decode the "LDPC source code". Further, considering that LDPC are very powerful codes, capable of performance very close to the theoretic limit in a lossless-channel DSC setup [14], the use of LDPC codes as

This material is based on research supported by the National Science Foundation under Grant No. CCF-0430634, and by the Commonwealth of Pennsylvania, Department of Community and Economic Development, through the Pennsylvania Infrastructure Technology Alliance (PITA).

the outer "source" code is particularly appealing. For the inner code, we focus on convolutional codes, since (i) they are soft-decodable, and (ii) they offer flexible code rates via puncturing, making it easy to adapt to the channel condition.

Following the discussion of individual sub encoders and sub decoders, we propose in Section III a joint iterative decoder that iterates soft information between component decoders. Unlike the conventional *channel SCC* decoder where the side information (SI), if any, is fed into the inner sub-decoder, with this *source-channel SCC* decoder, the SI will be provided to the outer sub-decoder, used directly for source decoding and indirectly (through decoding iterations) for channel decoding. Simulations show that considerable "joint decoding gain" can be achieved over a sequential decoding approach for the same complexity.

II. System Model

A. Preliminaries and System Model

Consider two i.i.d (independent and identically distributed) binary memoryless sources, X and Y, with output sequences $x_1, x_2, \cdots$ and $y_1, y_2, \cdots$. When the two sources are separately encoded and jointly decoded, the achievable rate region is bounded by the Slepian-Wolf boundary [1]:

$$R_x \geq H(X|Y), \quad R_y \geq H(Y|X), \quad R_x + R_y \geq H(X,Y). \quad (1)$$

Specifically, the corner points on the Slepian-Wolf boundary, known as asymmetric DSC, can be achieved by considering one source (e.g. Y) as the side information (i.e. compressed using a conventional entropy-achieving compression method) and compressing the other (i.e. X) to its conditional entropy ($H(X|Y)$). The line connecting the corner points can then be achieved through time-sharing.

It has been shown that the key to efficient DSC lies in powerful linear channel codes. Specifically, the binning approach [1] provides a general framework for asymmetric DSC to achieve a side compression rate of $n:(n-k)$ using an (n,k) linear code. The idea is to view length n source sequences X as virtual codewords, group them (2^n altogether) to 2^{n-k} bins, and to index the bins using 2^{n-k} syndrome sequences. By transmitting the length $n-k$ syndrome sequences instead of the original length n codewords, compression is achieved. To enable lossless recovery of the source sequences at the joint decoder, the following constraints need to be observed: (i) the rate of the channel code $k/n \leq 1-H(X|Y)$; (ii) a *geometrical uniformity* property needs to be preserved in each bin [1]; (iii) the channel code needs to be powerful enough to support the *virtual transmission channel* which is specified by the correlation between sources X and Y. Detailed discussion on the binning idea can be found in [1] [2].

In our approach, the source coding part will closely follow the binning idea. The following assumptions are used for the system setup: (i) X and Y are equiprobable memoryless binary random sources; (ii) X and Y are correlated at the same time instant: $Pr[X \neq Y] = p < 0.5.$; (iii) One of the sources, Y, is losslessly available at the joint decoder, and the other source, X, is to be compressed and transmitted through an additive white Gaussian noise (AWGN) channel. This "one-noisy-channel" setup is a special case of the general case where Y may also be corrupted. %enditemize

Since X and Y are correlated, let us treat Y (side information) as a distorted version of X. This is equivalent to modeling X and Y as the respective input and output of a binary symmetric channel (BSC) with crossover probability p. This thus results in a parallel-channel model for the noisy-channel DSC problem as shown in Fig. 1.

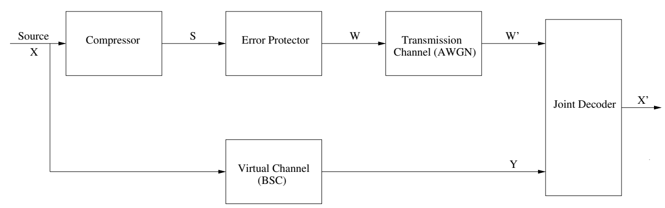

Fig. 1. A parallel-channel model for the one noisy-channel DSC problem.

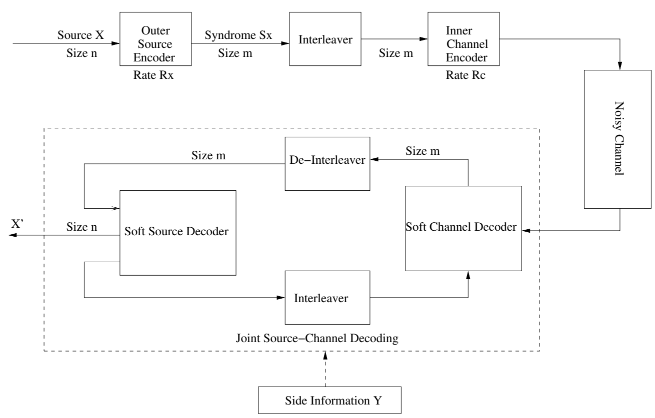

Fig. 2. System model for the SCC approach to the Slepian-Wolf Problem.

B. SCC approach to the Slepian-Wolf Problem

As mentioned earlier, we employ SCC codes in the asymmetric Slepian-Wolf compression with lossless side information (i.e. Y) available at the decoder. The system model is shown in Fig. 2. At the encoder, source sequences X will firstly be compressed to their syndrome sequences S_x at compression rate $R_x \geq H(X|Y)$ using the outer code. The syndrome sequences will then be error protected by the inner code before putting onto the noisy channel. The decoder employs an iterative process, where the extrinsic reliability information of S_x is iteratively exchanged between the two soft sub decoders. It should be noted that unlike conventional iterative decoders, here the SI Y is used to provide *a priori* information of X to the outer decoder rather than the inner decoder. In general, a random interleaver needs to reside between the outer and inner code to break up the correlation and to trigger a possible interleaving gain as shown in Fig. 2. When the outer code is an LDPC code, due to the randomness in its parity check matrix, this interleaver may not be needed..

III. Coding Procedure

The previous section has provided a general view of the SCC approach. This section discusses in detail the design and implementation of the LDPC-convolutional SCC scheme.

A. Encoding Procedure

An (n, k) low density parity check code is described either by an $m \times n$ parity-check matrix H (where $m = n - k$) or by its associated bipartite graph with m check nodes and n message nodes. A message node j is connected to a check node i if and only if the corresponding element (i, j) in H has a value of 1. The parity check matrix H usually contains randomly and sparsely distributed 1's, whose distribution is specified by the row/check degree polynomial $\rho(x)$ and column/bit degree polynomial $\lambda(x)$. If the degree distribution polynomials have only one term, then this LDPC code is considered regular; otherwise, it is irregular.

As discussed before, compression using channel codes is achieved by mapping long (codeword) sequences to short (syndrome) sequences. It is straight-forward to see that, for an (n, k) linear block code with a parity check matrix H, finding the length m syndrome sequence, S_x, for a length n message sequence, X, consists essentially of matrix multiplication of H with X. For an LDPC code represented in the bipartite graph form, this is equivalent to feeding information bits X at the message nodes, and computing the syndrome bits S_x at the check nodes using binary addition of all the message node values that are connected to the same check node:

$$s_j = \sum_{i \in \mathcal{C}_j}\!\!\oplus x_i, \quad j = 1, 2, \cdots, m, \tag{2}$$

where s_j denotes the jth syndrome bit, x_i the ith message bit, and C_j the set of message bits associated with the jth check. This procedure is illustrated in Fig. 3.

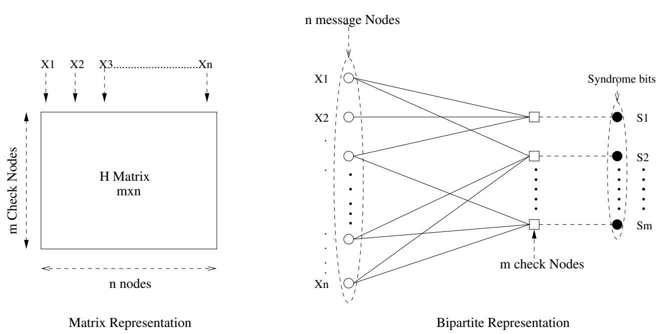

Fig. 3. LDPC source encoder (using the original H matrix).

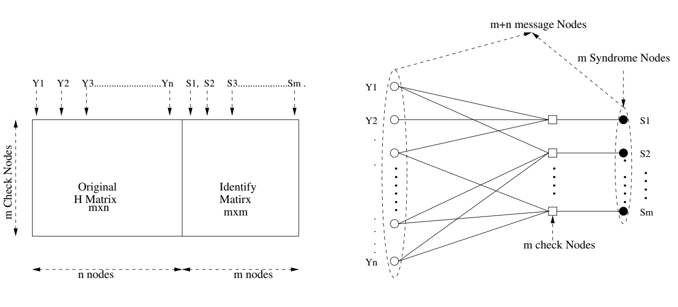

Fig. 4. LDPC source decoder (using the extended H matrix).

Since the syndrome sequence S_x is to be transmitted over a noisy channel, convolutional codes will be used to add redundancy bits for error resilience. The addition of the redundant bits will reduce the overall code/compression rate, but is a necessary step toward reliable transmission. Since a separate convolutional code is used for error protection, we have the luxury of adjusting the code rate (by puncturing) such that just-enough amount of redundancy is inserted for best channel utilization.

B. Decoding Procedure

The joint iterative decoder consists of two soft sub decoders corresponding to the inner convolutional code and the outer LDPC code. Below we discuss in detail how each individual sub decoder works and how soft messages are efficiently exchanged between them (Fig. 4 and 5).

We use log-likelihood ratio (LLR) values to represent the soft message. Notice that while there is global message exchange between the inner and outer sub decoders (Fig. 5), there is also local message exchange within the outer sub decoder (Fig. 4). For ease of proposition, let (g, l) denote the number of iterations, where g refers to the total number of global iterations, and l the total number of local iterations. Hence, an average of l/g rounds of message exchange is performed within the outer sub decoder before the message goes back to the inner sub decoder for refinement. It should be noted that subject to a constraint on the overall complexity, the number of local iterations vs the number of global iterations need to be carefully balanced. It should also be emphasized that no matter at what iteration and in which sub decoder, message exchange should always follow the turbo principle, that is, the outbound message to a processing unit should contain minimal correlation with the inbound message from that specific processing unit.

LDPC Source Decoder: We start with the LDPC sub decoder. It should have been clear from the LDPC source encoding procedure (see Eqn. (2)) that

$$s_j \oplus \sum_{i \in \mathcal{C}_j}\!\!\oplus x_i = 0, \quad j = 1, 2, \cdots, m. \tag{3}$$

That is, if m additional message nodes are introduced to pass the values of the syndrome bits to the associated check nodes, then the combination of the original n message bits and the m syndrome bits will have completed all checks. Equivalently, if we construct an extended parity check matrix H_{ext} by concatenating an $m \times m$ identity matrix to the original parity check matrix H as shown in Fig. 4, then we have $[X, S_x]$ as a valid codeword to this extended LDPC code (for convenience, we abuse the notation and assume both X and S_x are row vectors):

$$H_{ext} \times [X,\ S_x] = \mathbf{0}. \tag{4}$$

Hence, to recover the original data source X is to find a valid codeword $[X, S_x]$ from the noise-corrupted version $[Y \tilde{S}_x]$ for the extended LDPC code. This can be conveniently done by applying the message-passing algorithm to H_{ext}. We note that the same source decoding procedure is also used in [14] for noiseless-channel DSC. It should be noted that the *a priori* LLR information of S_x at the input of the message-passing decoder is obtained from the inner convolutional decoder, whereas that of X is computed from Y using the BSC correlation model:

$$L_{ap}(X) = (2Y - 1) \log \frac{p}{1-p}. \quad (5)$$

Upon each (local) message-passing iteration, the overall LLR information of X can be used to detect X, and the extrinsic LLR information of S_x (i.e. the overall LLR subtracts the *a priori* LLR) can be passed to the inner convolutional code.

The Convolutional Channel Decoder: The inner convolutional code uses a maximum *a posteriori* probability (APP) decoder or the BCJR algorithm to decode S_x. At the first global iteration, the input to the BCJR decoder consists only of AWGN-corrupted syndrome bits (from the channel). Starting from the second global iteration, feedback information from the outer LDPC code will be also be used as *a priori* information. The global message exchange is illustrated in Fig. 5.

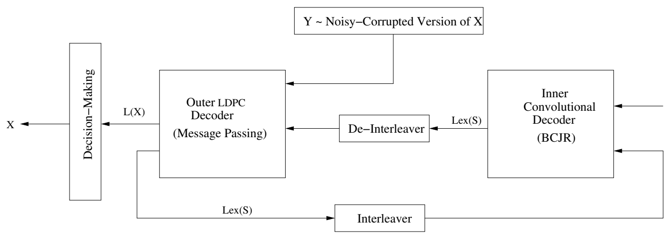

Fig. 5. Joint soft source-channel decoding.

IV. SIMULATION

This section evaluates the performance of the proposed scheme using computer simulations. Since the block size is typically limited to a few thousand bits in practical cases, we use $n = 2000$ bits as the length of the message sequences. It has been shown in [17] that at such short block sizes, regular LDPC codes tend to outperform irregular ones. Hence, regular LDPC codes with column weight 3 are used as the outer source code. A 16-state convolutional code with a generator polynomial $[1+D+D^4, 1+D^2+D^3+D^4]$ is used as the base channel code for error protection (can be punctured to other rates).

A. Compression Rate

One key advantage of the proposed scheme is that the compression rate and error protection capability can be independently tuned and controlled for any finite block size, whereas this is not possible with the joint source-channel DSC approaches in previous works (e.g. [10]-[12]). The first step in our system implementation is to determine the maximum compression rate that can be supported. Suppose that the inner error protection code is powerful enough to successfully recover the syndrome sequence S_x. Then, the equivalent channel as seen by the outer LDPC source code appears noiseless, and a noiseless-channel DSC setup can therefore be used to test and determine the rate of the LDPC code. Computer simulations show that an extended LDPC code based on an $(n, k) = (2000, 1000)$ regular LDPC code with $\lambda(x) = x^2$ and $\rho(x) = x^5$ can achieve a bit error rate (BER) of 10^{-6} when $p = 0.0533$. (Due to the space limitation, the performance plot is left out.) This suggests that a compression rate of no larger than 1/2 should be used for length 2000 source sequences withs a correlation factor $p = 0.0533$.

B. Error Resilient Capability

The next step is to determine how much redundancy is needed for satisfactory error protection of the compressed data (i.e. syndrome sequences). Again, this can be adjusted by using different inner codes and different code rates. Fig. 6 shows the performance when the inner convolutional code has rate $R_c = 3/4$ (punctured from the rate 1/2 mother code mentioned previously). If a distortion of around 10^{-6} is considered lossless, then this concatenated LDPC-convolutional scheme can achieve a compression rate of $R_1 R_c = \frac{2}{1} \frac{3}{4} = 3:2$ for source sequences having a length of 2000 bits and a correlation level $p = 0.0533$ on AWGN channels with SNR $= 1$ dB.

To provide a better compression rate for the same setup or to provide the same compression rate for a worse setup, either a stronger convolutional code, or a stronger LDPC code or a longer block size is needed. For example, by lowering the channel code rate to $R_c = 1/2$, an SNR of only -1 dB is needed for transmission over the noisy AWGN channel (Fig. 7). Alternatively, by increasing the sequence length to $n = 6000$ and subsequently using a $(6000, 3000)$ LDPC code, an correlation level of $p = 0.064$ can be supported with an AWGN SNR slight lower than 1 dB (plot not shown).

C. Joint Decoding Gain

From the above plots, we observe that the BER decreases with the increase of the number of the global iterations (up to a saturation point). This confirms that iterative interaction between the inner channel decoder and the outer source decoder improves the performance. Specifically, in Fig. 6, we have also plotted the performance of the sequential decoding where no feedback is provided from the outer source decoder back to the inner channel decoder. For the specific case that is shown (in dashed lines), one round of BCJR decoding of the inner channel code is performed, followed by 71 rounds of message-passing decoding of the outer source code. As will be clear from the complexity analysis in the next Subsection (Tab. I), this $(1, 71)$ sequential decoding scheme is of the same complexity as the $(6, 36)$ iterative decoding scheme, but of much worse performance especially at low SNRs. (The high error floor in the figure is in part due to the short code length.)

D. Complexity Analysis

In addition to adjusting the individual code rates and lengths to meet a performance requirement (i.e. guaranteed performance), the proposed scheme also makes it possible to, subject to a given complexity, fine-tune the performance by adopting different decoding scheduling schemes (i.e. best-effort performance). The latter can be facilitated by the complexity analysis.

Tab. I lists the complexity incurred in the inner BCJR decoder and the outer message-passing decoder (for the extended LDPC code). All computations are in the log domain, and table lookup is assumed for any nonlinear operation. In the table, M is used to denote the memory size of the convolutional code, g the number of global iterations, l the total number of local iterations in the LDPC decoder, m the length of the syndrome sequences of the LDPC code (i.e. the size of the compressed message), and λ and ρ the column weight and row weight of the (original) LDPC code. (Due to the space limitation, the actual computation is omitted.)

V. Conclusion and Future Works

We have proposed a novel LDPC-convolutional scheme for noisy-channel Slepian-Wolf coding. The key advantages include (i) *separate tuning and refining* of the compression rate (provided by the outer code) and the error protection power (provided by the inner code) is made possible and the overall performance is thus predictable and controllable, (ii) the use of the joint iterative decoder maximally exploits the performance without involving the tricky task of joint source-channel optimization, and (iii) many useful findings and results about serially concatenated codes can be borrowed, making system design and implementation a much lighter task.

The paper has also explored possible optimization to achieve better performance. We show that for short block sizes, a performance gain of 2.5 dB can be achieved using joint iterative decoding over separate decoding for the same decoding complexity, For large block sizes, irregular LDPC codes are known to perform better than regular LDPC codes. The application of irregular LDPC codes to our system model and its effect on the performance of the system will be further studied. It is interesting to note that when an outer irregular LDPC code is used and when the inner convolutional code is limited to an accumulator, the resulting serial concatenation essentially becomes a systematic IRA code [15]. Additionally, the above work can be extended to a case where the side information is not perfect.

TABLE I

Complexity analysis of log-MAP and MP

	log-MAP	MP
$\max$ ops	$gm(5 \cdot 2^M - 2)$	
additions	$gm(15 \cdot 2^M + 9)$	$((2\lambda)n + (2\rho - 1)m)l$
look-ups	$gm(5 \cdot 2^M - 2)$	$(2\rho - 2)l + mg$

References

[1] D. Slepian and J.K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol.19,pp.471-480, July 1973.

[2] R. Zamir, S. Shamai, and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Trans. Inform. Theory*, pp. 1250-1276, June 2002.

[3] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): Design and construction," *Proc. of Data Compression Conference (DCC)*, pp.158-167, Mar. 1999.

[4] S. S. Pradhan and K. Ramchandran, "Distributed source coding: Symmetric rates and applications to sensor networks," *Proc. of Data Compression Conference (DCC)*, pp. 363-372, Mar. 2000.

[5] J.Bajcsy and Patrick Mitran, "Coding for the Slepian-Wolf problem with turbo codes," *Proc. IEEE GLOBECOM*, vol.2, pp.1400-1404, Nov. 2001.

[6] J. Garcia-Frias and Y.Zhao, "Compression of correlated binary sources using turbo codes," *IEEE Comm. Letters*, vol.5, pp.417-419, Oct2001.

[7] A. Aaron and B.Girod, "Compression of correlated binary sources using turbo codes," *Proc. of Data Compression Conference (DCC)*, pp.252-261, Apr. 2002.

[8] A.D. Liveris, Z. Xiong and C.N. Georghiades, "A distributed source coding technique for highly correlated images using turbo-codes," *Proc. IEEE ICASSP*, vol. 4, pp. 3261-3264, May 2002.

[9] Z. Tu, J. Li, and R. S. Blum, "Compression of a binary source with side information using parallel concatenated convolutional codes," to appear *IEEE GLOBECOM*, Dallas, TX, 2004.

[10] J. Garcia-Frias and Y.Zhao, "Joint source-channel decoding of correlated sources over noisy channels," *Proc. IEEE DCC*, pp. 283-292, Mar 2001.

[11] P. Mitran and J.Bajcsy, "Turbo source coding: A noise-robust approach to data compression," *Proc. IEEE DCC*, pp.465, Apr. 2002.

[12] G.-C. Zhu, and F. Alajaji, "Turbo codes for nonuniform memoryless sources over noisy channels," *IEEE Comm. Letters*, vol. 6, pp. 64-66, Feb 2002.

[13] T. Murayama, "Statistical mechanics of linear compression codes in network communication," *Europhysics Letters*, preprint, 2001.

[14] A. D. Liveris, Z. Xiong and C. N. Geoghiades, "Compression of binary sources with side information using low-density parity-check Codes," 2002.

[15] A. D. Liveris, Z. Xiong and C. N. Georghiades, "Joint source-channel coding of binary sources with side information at the decoder using IRA codes," *Multimedia Signal Processing Workshop*, Dec.2002.

[16] S. Shamai, S. Verdu, and R. Zamir, "Systematic lossy source/channel coding", *IEEE Trans. Inform. Theory*, vol. 44, pp. 564-579, March 1998.

[17] D. J. C. MacKay and M. C. Davey, "Evaluation of Gallager codes for short block length and high rate applications," *Proc. IMA Workshop on Codes, Systems and Graphical Models*, pp. 113-130, 1999.

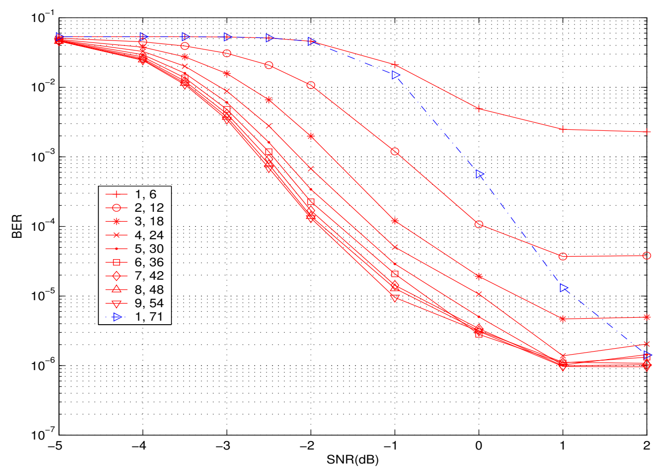

Fig. 6. Performance of the proposed LDPC-convolutional scheme for noisy-channel DSC problem. ("g, l" in the legend refers to the total number of global iterations and local iterations; $p = 0.0533$; $(2000, 1000)$ LDPC code followed by $(1333, 1000)$ convolutional code; overall compression rate $3:2$.

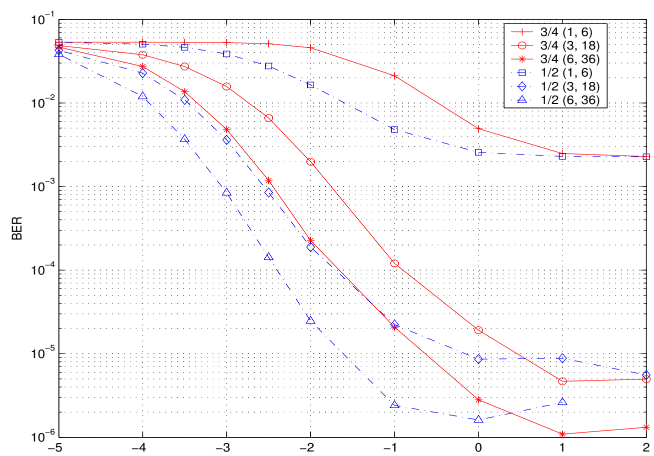

Fig. 7. System performance for different convolutional code rates. ($p = 0.0533$; $(2000, 1000)$ LDPC code; convolutional code rate $R_c = 1/2, 3/4$)