

RESEARCH

Open Access

# Distributed joint source-channel coding for relay systems exploiting source-relay correlation and source memory

Xiaobo Zhou<sup>1\*</sup>, Meng Cheng<sup>1</sup>, Khoirul Anwar<sup>1</sup> and Tad Matsumoto<sup>1,2</sup>

## Abstract

In this article, we propose a distributed joint source-channel coding (DJSCC) technique that well exploits source-relay correlation as well as source memory structure simultaneously for transmitting binary Markov sources in a one-way relay system. The relay only *extracts* and forwards the source message to the destination, which implies imperfect decoding at the relay. The probability of errors occurring in the source-relay link can be regarded as source-relay correlation. The source-relay correlation can be estimated at the destination node and utilized in the iterative processing. In addition, the memory structure of the Markov source is also utilized at the destination. A modified version of the Bahl, Cocke, Jelinek, and Raviv (BCJR) algorithm is derived to exploit the memory structure of the Markov source. Extrinsic information transfer (EXIT) chart analysis is then performed to investigate convergence property of the proposed technique. Results of simulations conducted to evaluate the bit-error-rate (BER) performance and the EXIT chart analysis show that, by exploiting the source-relay correlation and source memory simultaneously, our proposed technique achieves significant performance gain, compared with the case where the correlation knowledge is not fully used.

## Introduction

Wireless mesh and/or sensor networks having great number of low-power consuming wireless nodes (e.g., small relays and/or micro cameras) have attracted a lot of attention of the society, and a variety of its potential applications has been considered recently [1]. The fundamental challenge of wireless mesh and/or sensor networks is how energy-/spectrum-efficiently as well as reliably the multiple sources can transmit their originating information to the multiple destinations. However, such multi-terminal systems have two practical limitations: (1) wireless channel suffers from various impairments, such as interference, distortions and/or deep fading, (2) signal processing complexity as well as transmitting powers has to be as low as possible due to the power, bandwidth, and/or size restrictions of the wireless nodes.

Cooperative communication techniques provide a potential solution to the problems described above, due

to its excellent transmit diversity for fading mitigation [2]. One simple form of cooperative wireless communications is a single relay system, which consists of one source, one relay and one destination. The role of the relay is to provide alternative communication route for transmission, hence improving the probability of successful signal reception of source information sequence at the destination. In this relay system, the information sent from the source and the relay nodes are correlated, which in this article is referred to as source-relay correlation. Furthermore, the information collected at the source node contains memory structure, according to the dynamics that governs the temporal behavior of the originator (or sensing target). The source-relay correlation and the memory structure of the transmitted data can be regarded as redundant information which can be used for source compression and/or error correction in distributed joint source-channel coding (DJSCC).

There are many excellent coding schemes which can achieve efficient node cooperative communications, such as [3,4], where decode-and-forward (DF) relay strategy is adopted and the source-relay link is assumed to be error

\*Correspondence: xiaobo@jaist.ac.jp

<sup>1</sup> School of Information Science, Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan

Full list of author information is available at the end of the article

free. In practice, when the signal-to-noise ratio (SNR) of the source-relay link falls below certain threshold, successful decoding at relay may become impossible. Besides, to completely correct the errors at the relay, strong codes such as turbo codes or low density parity check (LDPC) codes with iterative decoding are required, which will impose heavy computational burden at the relay. As a result, several coding strategies assuming that the relay cannot always decode correctly the information from the source have been presented in [5-7].

Joint source-channel coding (JSCC) has been widely used to exploit the memory structure inherent within the source information sequence. In the majority of the approaches to JSCC design, variable-length code (VLC) is employed as source encoder and the implicit residual redundancy after source encoding is additionally used for error correction in the decoding process. Some related study can be found in [8-11]. Also, there are some literatures which focus on exploiting the memory structure of the source directly, e.g., some approaches of combining hidden Markov Model (HMM) or Markov chain (MC) with the turbo code design framework are presented in [12-14].

In the schemes mentioned above, the exploitation of the source-relay correlation and the source memory structure have been addressed separately. Not much attention has been paid to relay systems exploiting the source-relay correlation and the source memory simultaneously. A similar study can be found in [15], where the memory structure of the source is represented by a very simple model, bit-flipping between the current information sequence and its previous counterpart, which is not reasonable in many practical scenarios. When the exploitation of the source memory having more generic structures, the problem of code design for relay systems exploiting jointly the source-relay correlation and the source memory structure is still open.

In this article, we propose a new DJSCC scheme for transmitting binary Markov source in a one-way single relay system, based on [7,14]. The proposed technique makes efficient utilization of the source-relay correlation as well as the source memory structure simultaneously to achieve additional coding gain. The rest of this article is organized as follow. Section 'System model' introduces the system model. The proposed decoding algorithm is described in Section 'Proposed decoding scheme.' Section 'EXIT chart analysis' shows the results of extrinsic information transfer (EXIT) chart analysis conducted to evaluate the convergence property of the proposed system. Section 'Convergence analysis and BER performance evaluation' shows the bit-error-rate (BER) performance of the system based on EXIT chart analysis. The simulation results for image transmission using the proposed technique is presented in Section 'Application to image

transmission'. Finally, conclusions are drawn in Section 'Conclusion' with some remarks.

## System model

### One-way single relay system

In this article, a single-source single-relay system is considered where all links are assumed to suffer from Additive White Gaussian Noise (AWGN). The relay system operates in a half-duplex mode. During the first time interval, the source node broadcasts the signal to both the relay and destination nodes. After receiving signals from the source, the relay *extracts* the data even though it may contain some errors, re-encodes, and then transmits the extracted data to the destination node in the second time interval.

The relay can be located closer to the source or to the destination, or the three nodes keep the same distance with each other. All these three different relay location scenarios are considered in this article, as shown in Figure 1. The geometric-gain [4]  $G_{xy}$  of the link between the node  $x$  and  $y$  can be defined as

$$G_{xy} = \left(\frac{d_{sd}}{d_{xy}}\right)^l, \quad (1)$$

where  $d_{xy}$  denotes the distance of the link between the node  $x$  and  $y$ . The pass loss exponent  $l$  is empirically set at 3.52 [4]. Note that the geometric-gain of the source-destination link  $G_{sd}$  is normalized to 1 without the loss of generality.

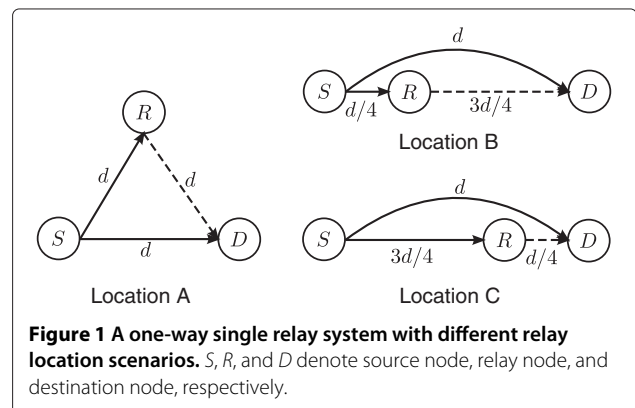
The received signals at the relay and at the destination nodes can be expressed as

$$\mathbf{y}_{sr} = \sqrt{G_{sr}} \cdot \mathbf{x} + \mathbf{n}_r, \quad (2)$$

$$\mathbf{y}_{sd} = \sqrt{G_{sd}} \cdot \mathbf{x} + \mathbf{n}_d, \quad (3)$$

$$\mathbf{y}_{rd} = \sqrt{G_{rd}} \cdot \mathbf{x}_r + \mathbf{n}_d, \quad (4)$$

where  $\mathbf{x}$  and  $\mathbf{x}_r$  represent the symbol vectors transmitted from the source and the relay, respectively. Notations



$\mathbf{n}_r$  and  $\mathbf{n}_d$  represent the zero-mean AWGN noise vectors at the relay and the destination with variances  $\sigma_r^2$  and  $\sigma_d^2$ , respectively. The SNR of the source-relay and relay-destination links with the three different relay location scenarios, as shown in Figure 1, can be decided as: for location A,  $\text{SNR}_{sr} = \text{SNR}_{rd} = \text{SNR}_{sd}$ ; for location B,  $\text{SNR}_{sr} = \text{SNR}_{sd} + 21.19$  dB and  $\text{SNR}_{rd} = \text{SNR}_{sd} + 4.4$  dB; for location C,  $\text{SNR}_{sr} = \text{SNR}_{sd} + 4.4$  dB and  $\text{SNR}_{rd} = \text{SNR}_{sd} + 21.19$  dB.

### Source-relay correlation

The diagram of the proposed relay strategy is illustrated in Figure 2. At the source node, the original information bits vector  $\mathbf{u}$  is first encoded by a recursive systematic convolutional (RSC) code, interleaved by  $\Pi_s$ , encoded by a doped accumulator (ACC) with a doping rate  $K_s$  [16] and then modulated using binary-phase shift keying (BPSK) to obtain the coded sequence  $\mathbf{x}$ . After obtaining the received signal  $\mathbf{y}_{sr}$  from the source, the relay performs the decoding process only once (i.e., no iterative processing at the relay) to retrieve  $\mathbf{u}_r$ , which is used as an estimate of  $\mathbf{u}$ .  $\mathbf{u}_r$  is first interleaved by  $\Pi_0$  and then encoded following the same encoding process as in the originating node with a doping rate  $K_r$  to generate the coded sequence  $\mathbf{x}_r$ .

Errors may occur between  $\mathbf{u}$  and  $\mathbf{u}_r$ , as shown in Figure 3, compared to the cases where iterative decoding is performed at the relay node. Apparently, with more iterations better BER performances can be achieved at the relay node. However, this advantage becomes negligible in low  $\text{SNR}_{sr}$  scenarios. Instead, the estimate of the source information sequence is simply *extracted* by performing the corresponding channel decoding process just once. Consequently, the relay complexity can be significantly reduced without causing any significant performance degradation by the proposed algorithm, as detailed in Section 'Proposed decoding scheme'.

The source-relay correlation indicates the correlation between  $\mathbf{u}$  and  $\mathbf{u}_r$ , which can be represented by a bit-flipping model, as shown in Figure 2.  $\mathbf{u}_r$  can be defined as  $\mathbf{u}_r = \mathbf{u} \oplus \mathbf{e}$ , where  $\mathbf{e}$  is an independent binary random variable and  $\oplus$  indicates modulus-2 addition. The correlation between  $\mathbf{u}$  and  $\mathbf{u}_r$  is characterized by  $p_e$ , where  $p_e = \Pr(\mathbf{e} = 1) = \Pr(\mathbf{u} \neq \mathbf{u}_r)$  [6].

### Markov source

In this article, the source we considered is a stationary state emitting binary Markov source  $\mathbf{u} = u_1 u_2 \dots u_t \dots$ , of which the transition matrix is:

$$A = [a_{ij}] = \begin{bmatrix} a_{0,0} & a_{0,1} \\ a_{1,0} & a_{1,1} \end{bmatrix} = \begin{bmatrix} p_1 & 1-p_1 \\ 1-p_2 & p_2 \end{bmatrix}, \quad (5)$$

where  $a_{ij}$  is the transition probability defined by

$$a_{ij} = \Pr\{u_t = j | u_{t-1} = i\}, \quad i, j = 0, 1. \quad (6)$$

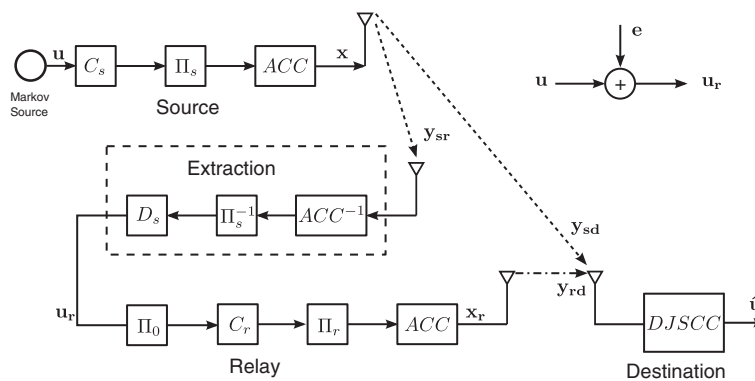
The entropy rate of stationary Markov source [17] is given by

$$H(S) = - \sum_{ij \in \{0,1\}} \mu_i a_{ij} \log a_{ij}, \quad (7)$$

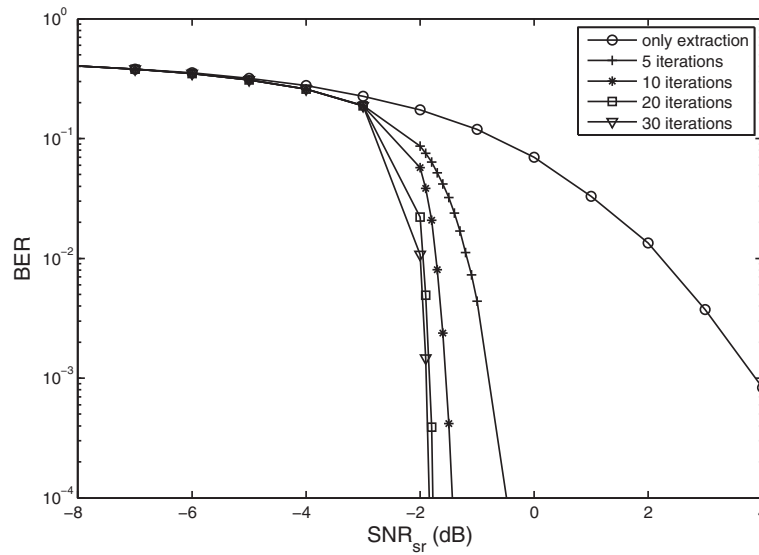
where  $\{\mu_i\}$  is the stationary state probability. The memory structure of Markov source can be characterized by the state transition probabilities  $p_1$  and  $p_2$ ,  $0 < p_1, p_2 < 1$ , with which  $p_1 = p_2 = 0.5$  indicates the memoryless source, while  $p_1 \neq 0.5$  or  $p_2 \neq 0.5$ , and hence  $H(S) < 1$  indicate source with memory.

### Proposed decoding scheme

The block diagram of the proposed DJSCC decoder for one-way relay system exploiting the source-relay



**Figure 2 Proposed relay strategy and its equivalent bit-flipping model.**  $C_s$  and  $C_r$  are RSC code,  $\Pi_s$  and  $\Pi_r$  are random interleavers. ACC and  $\text{ACC}^{-1}$  denote doped accumulator and decoder of the doped accumulator, respectively.



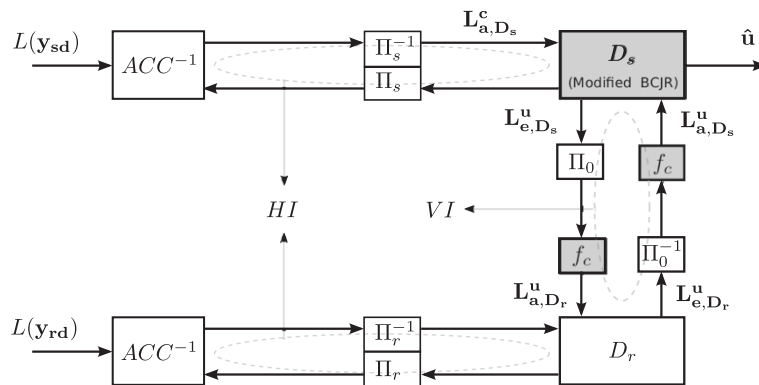
**Figure 3** BER of the source-relay link over AWGN channel versus  $\text{SNR}_{sr}$ . The doping rate at the source node is  $K_s = 1$ .

correlation and the source memory structure is illustrated in Figure 4. The maximum a posteriori (MAP) algorithm for the convolutional code proposed by Bahl, Cocke, Jelinek and Raviv (BCJR), is used for MAP-decoding of convolutional code and ACC. Here,  $D_s$  and  $D_r$  denote the decoder of  $C_s$  and  $C_r$ , respectively. In order to exploit the knowledge of the memory structure of the Markov source, the source and  $C_s$  are treated as a single constituent code. Hence, it is reasonable to represent the code structure by a super trellis by combining the trellis diagram of the source and  $C_s$ . A modified version of the BCJR algorithm is derived to jointly perform source and channel decoding over this super trellis at  $D_s$ . However,  $D_r$  cannot exploit the source memory due to the additional interleaver  $\Pi_0$ , as shown in Figure 2.

At the destination node, the received signals from the source and the relay are first converted to *log-likelihood ratio* (LLR) sequences  $L(y_{sd})$ ,  $L(y_{rd})$ , respectively, and then decoded via two horizontal iterations (HI), as shown in Figure 4. Then the extrinsic LLRs generated from  $D_s$  and  $D_r$  in the two HIs are further exchanged by several vertical iterations (VI) through an LLR updating function  $f_c$ , of which role is detailed in the following section. This process is performed iteratively, until the convergence point is reached. Finally, hard decision is made based on the a posteriori LLRs obtained from  $D_s$ .

#### LLR updating function

First of all, the correlation property (error probability occurring in the source-relay link)  $p_e$  is estimated at the



**Figure 4** The proposed DJSCC decoder for single relay system exploiting the source-relay correlation and the source memory structure.  $\text{ACC}^{-1}$  denotes the decoder of the doped accumulator.  $D_s$  and  $D_r$  denote the decoder of  $C_s$  and  $C_r$ , respectively.

destination using the a posteriori *LLRs* of the uncoded bits,  $L_{p,D_s}^u$  and  $L_{p,D_r}^u$  from the decoders  $D_s$  and  $D_r$ , as

$$\hat{p}_e = \frac{1}{N} \sum_{n=1}^N \frac{\exp(L_{p,D_s}^u) + \exp(L_{p,D_r}^u)}{[1 + \exp(L_{p,D_s}^u)] \cdot [1 + \exp(L_{p,D_r}^u)]}, \quad (8)$$

where  $N$  indicates the number of the a posteriori *LLR* pairs from the two decoders with sufficient reliability. Only the *LLRs* with their absolute values larger than a given threshold can be used in calculating  $\hat{p}_e$ .

After obtaining the estimated error probability using (8), the probability of  $\mathbf{u}$  can be updated from  $\mathbf{u}_r$  as

$$\Pr(u^k = 0) = (1 - \hat{p}_e) \cdot \Pr(u_r^k = 0) + \hat{p}_e \cdot \Pr(u_r^k = 1), \quad (9)$$

$$\Pr(u^k = 1) = (1 - \hat{p}_e) \cdot \Pr(u_r^k = 1) + \hat{p}_e \cdot \Pr(u_r^k = 0), \quad (10)$$

where  $u^k$  and  $u_r^k$  denote the  $k$ th elements of  $\mathbf{u}$  and  $\mathbf{u}_r$ , respectively. This leads to the *LLR* updating function [6] for  $\mathbf{u}$ :

$$L(u^k) = \ln \frac{\Pr(u^k = 0)}{\Pr(u^k = 1)} = \ln \frac{(1 - \hat{p}_e) \cdot \exp[L(u_r^k)] + \hat{p}_e}{(1 - \hat{p}_e) + \hat{p}_e \cdot \exp[L(u_r^k)]}. \quad (11)$$

Similarly, the *LLR* updating function for  $\mathbf{u}_r$  can be expressed as:

$$L(u_r^k) = \ln \frac{\Pr(u_r^k = 0)}{\Pr(u_r^k = 1)} = \ln \frac{(1 - \hat{p}_e) \cdot \exp[L(u^k)] + \hat{p}_e}{(1 - \hat{p}_e) + \hat{p}_e \cdot \exp[L(u^k)]}. \quad (12)$$

In summary, the general form of *LLR* updating function  $f_c$ , as shown in Figure 4, is given as

$$f_c(x) = \ln \frac{(1 - \hat{p}_e) \cdot \exp(x) + \hat{p}_e}{(1 - \hat{p}_e) + \hat{p}_e \cdot \exp(x)}, \quad (13)$$

where  $x$  denotes the input *LLRs*. The output of  $f_c$  is the updated *LLRs* by exploiting  $\hat{p}_e$  as the source-relay correlation. The *VI* operations of the proposed decoder can be expressed as

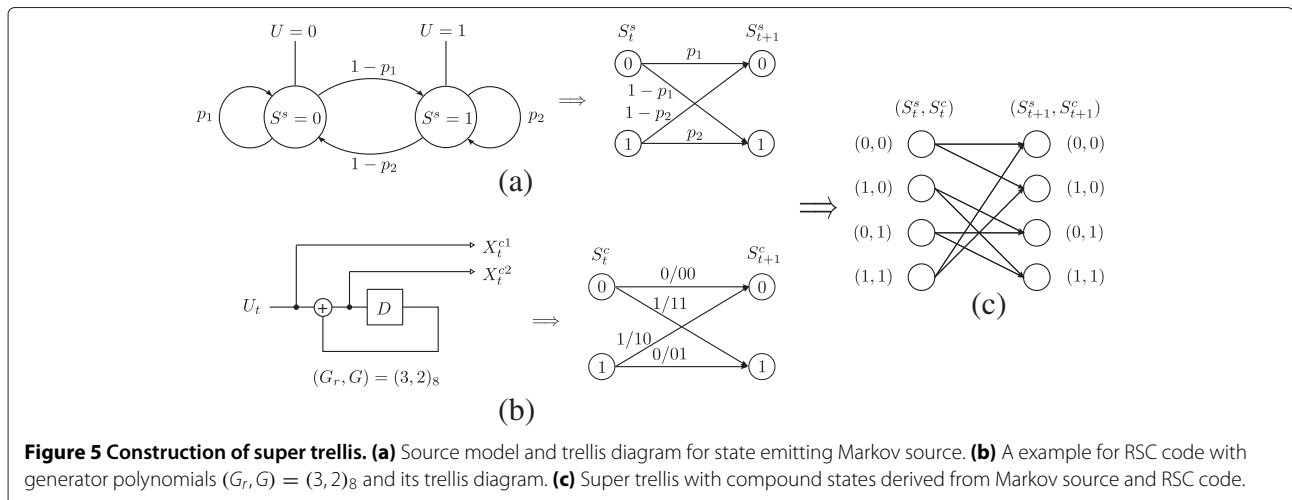
$$\mathbf{L}_{a,D_s}^u = f_c[\Pi_0^{-1}(\mathbf{L}_{e,D_r}^u)], \quad (14)$$

$$\mathbf{L}_{a,D_r}^u = f_c[\Pi_0(\mathbf{L}_{e,D_s}^u)], \quad (15)$$

where  $\Pi_0(\cdot)$  and  $\Pi_0^{-1}(\cdot)$  denote interleaving and de-interleaving functions corresponding to  $\Pi_0$ , respectively.  $\mathbf{L}_{a,D_s}^u$  and  $\mathbf{L}_{e,D_s}^u$  denote the a priori *LLRs* fed into, and extrinsic *LLRs* generated by  $D_s$ , respectively, both for the uncoded bits. Similar definitions should apply to  $\mathbf{L}_{a,D_r}^u$  and  $\mathbf{L}_{e,D_r}^u$  for  $D_r$ .

#### Joint decoding of Markov source and channel encoder $C_s$ Representation of super trellis

Assume that the  $C_s$  is a memory length  $\nu$  convolutional code. There are  $2^\nu$  states in the trellis diagram of this code, which are indexed by  $m$ ,  $m = 0, 1, \dots, 2^\nu - 1$ . The state of  $C_s$  at the time index  $t$  is denoted as  $S_t^c$ . Similarly, there are two states in order-1 binary Markov source, and the state at the time index  $t$  is denoted as  $S_t^s$  with  $S_t^s \in \{0, 1\}$ . For a binary Markov model described in Section 'System model', the source model and its corresponding trellis diagram are illustrated in Figure 5a. The output value at a time instant  $t$  from the source is the same as the state value of  $S_t^s$ . The trellis branches represent the state transition of which probabilities have been defined by (6). On the other hand, for  $C_s$ , the branches in its trellis diagram indicate input/output characteristics.



**Figure 5 Construction of super trellis. (a)** Source model and trellis diagram for state emitting Markov source. **(b)** A example for RSC code with generator polynomials  $(G_r, G) = (3, 2)_8$  and its trellis diagram. **(c)** Super trellis with compound states derived from Markov source and RSC code.

At time instant  $t$ , the state of the source and the state of the  $C_s$  can be regarded as a new state  $(S_t^s, S_t^c)$ , which leads to the super trellis diagram. A simple example of combining binary Markov source with a recursive convolutional code (RSC) with generator polynomial  $(G_r, G) = (3, 2)_8$  is depicted in Figure 5. At each state  $(S_t^s, S_t^c)$ , the input to the outer encoder is determined, given the state of the Markov source. Actually, the new trellis branches can be regarded as a combination corresponding to the branches of the Markov source and of the trellis diagram of  $C_s$ . Hence, the new trellis branches represent both state transition probabilities of the Markov source and input/output characteristics of  $C_s$  defined in its trellis diagram.

It should be noticed that a drawback of this approach is the exponentially growing number of the states in the super trellis. However, if  $C_s$  is only a short memory convolutional code, the complexity increase is due mainly to the number of Markov source states. In fact, it is shown in Section 'Convergence analysis and BER performance evaluation' that, even with a memory-1 code used as  $C_s$  can achieve excellent performance. Therefore, the complexity is largely the issue of source modeling depending on applications.

#### Modified BCJR algorithm for super trellis

In this section, we make modifications of the standard BCJR algorithm [18] for the decoding performed over the super trellis constructed in the previous section. Here, we ignore momentarily the serially concatenated structure, and only focus on the decoding process performed over the super trellis diagram. For a convolutional code with memory length  $\nu$ , there are  $2^\nu$  states in its trellis diagram, which is indexed by  $m$ ,  $m = 0, 1, \dots, 2^\nu - 1$ . The input sequence to the encoder  $\mathbf{u} = u_1 u_2 \dots u_t \dots u_L$ , which is also a series of the states of Markov source, is assumed to have length  $L$ . The output of the encoder is denoted as  $\mathbf{x} = \{\mathbf{x}^1, \mathbf{x}^2\}$ . The coded binary sequence is BPSK mapped and then transmitted over AWGN channels. The received signal is a noise-corrupted version of the BPSK mapped sequence, denoted as  $\mathbf{y} = \{\mathbf{y}^1, \mathbf{y}^2\}$ . The received sequence from the time indexes  $t_1$  to  $t_2$  is denoted as  $\mathbf{y}_{t_1}^{t_2} = \mathbf{y}_{t_1}, \mathbf{y}_{t_1+1}, \dots, \mathbf{y}_{t_2}$ .

The aim of the modified BCJR algorithm is to calculate conditional *log-likelihood ratio* (LLR) of the coded bits  $\{x_t^{c1}\}$ , based on the whole received sequence  $\mathbf{y}_1^L$ , which is defined by

$$L(x_t^{c1}) = \ln \frac{\Pr(x_t^{c1} = 1 | \mathbf{y}_1^L)}{\Pr(x_t^{c1} = 0 | \mathbf{y}_1^L)} = \ln \frac{\sum_{(i,m) \in B_t^1} \Pr(S_t^s = i, S_t^c = m, \mathbf{y}_1^L)}{\sum_{(i,m) \in B_t^0} \Pr(S_t^s = i, S_t^c = m, \mathbf{y}_1^L)}, \quad (16)$$

where  $B_t^k$  denotes the sets of states  $\{(S_t^s = i, S_t^c = m)\}$  yielding the systematic output  $x_t^{c1}$  of the  $C_s$  being  $k$ ,  $k = 0, 1$ .

In order to compute the last term in (16), three parameters indicating the probabilities defined as below have to be introduced:

$$\alpha_t(i, m) = \Pr(u_t = i, S_t = m, \mathbf{y}_1^t), \quad (17)$$

$$\beta_t(i, m) = \Pr(\mathbf{y}_{t+1}^L | u_t = i, S_t = m), \quad (18)$$

$$\gamma_t(\mathbf{y}_t, i', m', i, m) = \Pr(u_t = i, S_t = m, \mathbf{y}_t | u_{t-1} = i', S_{t-1} = m'). \quad (19)$$

Now we have

$$\Pr(S_t^s = i, S_t^c = m, \mathbf{y}_1^L) = \alpha_t(i, m) \beta_t(i, m). \quad (20)$$

Substituting (20) in (16), we obtain the whole set of equations for the modified BCJR algorithm.  $\alpha_t(i, m)$ ,  $\beta_t(i, m)$ ,  $\gamma_t(\mathbf{y}_t, i', m', i, m)$  are found to be functions of both the output of Markov source and the states in the trellis diagram of  $C_s$ . More specifically,  $\gamma_t(\mathbf{y}_t, i', m', i, m)$  represents information of input/output relationship corresponding to the state transition  $S_t = m' \rightarrow S_t = m$ , specified by the trellis diagram of  $C_s$ , as well as of the state transition probabilities depending on Markov source. Therefore,  $\gamma$  can be decomposed as

$$\gamma_t(\mathbf{y}_t, i', m', i, m) = \begin{cases} a_{i',i} \cdot \gamma_t^*(\mathbf{y}_t, m', m), & \text{if } (i', m') \in E_t(i, m); \\ 0, & \text{otherwise,} \end{cases} \quad (21)$$

where  $a_{i',i}$  is defined in (6), and  $\gamma_t^*(\mathbf{y}_t, m', m)$  is defined as

$$\gamma_t^*(\mathbf{y}_t, m', m) = \Pr(\mathbf{y}_t, S_t = m | S_{t-1} = m'). \quad (22)$$

$E_t(i, m)$  is the set of states  $\{(u_{t-1}, S_{t-1})\}$  that have a trellis branch connected with state  $(u_t = i, S_t = m)$  in the super trellis.

After  $\gamma$  is obtained,  $\alpha$  and  $\beta$  can also be computed via the following recursive formulae

$$\alpha_t(i, m) = \sum_{i', m'} \alpha_{t-1}(i', m') \gamma_t(\mathbf{y}_t, i', m', i, m), \quad (23)$$

$$\beta_t(i, m) = \sum_{i', m'} \gamma_t(\mathbf{y}_t, i', m', i, m) \beta_{t+1}(i', m'). \quad (24)$$

Since the output encoder always starts from the state zero, while the probabilities for the Markov source starts from state "0" or state "1" is equal. Hence, the appropriate boundary condition for  $\alpha$  is  $\alpha_0(0, 0) = \alpha_0(1, 0) = 1/2$  and  $\alpha_0(i, m) = 0$ ,  $i = 0, 1; m \neq 0$ . Similarly, the boundary conditions for  $\beta$  is  $\beta_L(i, m) = 1/2^{\nu+1}$ ,  $i = 0, 1; m = 0, 1, \dots, 2^\nu - 1$ .

Now the whole set of equations for the modified BCJR algorithm can be obtained. Combining all the results



described above, we can obtain the conditional LLRs for  $x_t^{c1}$ , as

$$L(x_t^{c1}) = L_{ap}(x_t^{c1}) + L_{ch}(x_t^{c1}) + L_{ex}(x_t^{c1}), \quad (25)$$

where

$$L_{ap}(x_t^{c1}) = \ln \frac{\Pr(x_t^{c1} = 1)}{\Pr(x_t^{c1} = 0)}, \quad (26)$$

$$L_{ch}(x_t^{c1}) = \ln \frac{\Pr(y_t^{c1} | x_t^{c1} = 1)}{\Pr(y_t^{c1} | x_t^{c1} = 0)}, \quad (27)$$

$$L_{ex}(x_t^{c1}) = \ln \frac{\sum_{(i,m) \in B_t^1} \sum_{i',m'} \alpha_{t-1}(i',m') \gamma_t(y_t^{c2}, i', m', i, m) \beta_t(i, m)}{\sum_{(i,m) \in B_t^0} \sum_{i',m'} \alpha_{t-1}(i',m') \gamma_t(y_t^{c2}, i', m', i, m) \beta_t(i, m)} \quad (28)$$

representing the a priori LLR, the channel LLR and the extrinsic LLR, respectively. The same representation should apply to  $\{x_t^{c2}\}$ .

### EXIT chart analysis

In this section, we present results of three-dimensional (3D) EXIT chart [19-21] analysis conducted to identify the impact of the memory structure of the Markov source and the source-relay correlation on the joint decoder. The analysis focuses on the decoder  $D_s$  since the main aim is to successfully retrieve the information estimates  $\hat{u}$ . As shown in Figure 4, the decoder  $D_s$  exploits two a priori

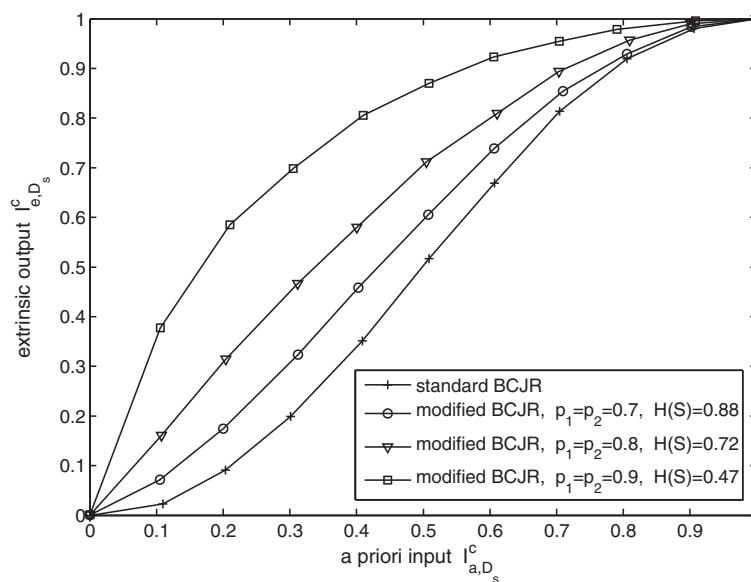
LLRs:  $L_{a,D_s}^c$  and the updated version of  $L_{e,D_r}^u, L_{a,D_s}^u$ . Therefore, the EXIT function of  $D_s$  can be characterized as

$$I_{e,D_s}^c = T_{D_s}^c(I_{a,D_s}^c, I_{e,D_r}^u, p_e), \quad (29)$$

where  $I_{e,D_s}^c$  denotes the mutual information between the extrinsic LLRs,  $L_{e,D_s}^c$  generated from  $D_s$ , and the coded bits of  $D_s$ .  $I_{e,D_s}^c$  can be obtained by the histogram measurement [21]. Similar definitions can be applied to  $I_{a,D_s}^c$  and  $I_{e,D_r}^u$ .

The second parameter of  $T_{D_s}^c(\cdot)$ ,  $I_{e,D_r}^u$ , represents extrinsic information generated from source-relay correlation. Meanwhile, the modified BCJR algorithm adopted by  $D_s$  utilizes the memory structure of Markov source. At first, we assume that the source-relay correlation is not exploited and only focus on the exploitation of source memory. In this case,  $I_{e,D_r}^u = 0$  and the EXIT analysis of  $D_s$  can be simplified to two-dimensional. The EXIT curves with and without the modifications described in the previous section are illustrated in Figure 6. The code used in the analysis is a half rate memory-1 RSC with the generator polynomials  $(G_r, G) = (3, 2)_8$ . It can be observed from Figure 6 that, compared to the standard BCJR algorithm, the EXIT curves obtained by using the modified BCJR algorithm are lifted up over the whole a priori input region, indicating that larger extrinsic information can be obtained. It is also worth noticing that the contribution of source memory represented by the increase in extrinsic mutual information becomes larger as the entropy of Markov source decreases.

Next we conducted 3D EXIT chart analysis for  $D_s$  to evaluate the impact of the source-relay correlation,



**Figure 6** Extrinsic information transfer characteristic of  $D_s$ , with standard BCJR and with modified BCJR. The source-relay correlation is not considered. Generator polynomials of  $C_s$  is  $(G_r, G) = (3, 2)_8$ .

where the source memory is not exploited. The corresponding EXIT planes of  $D_s$ , shown in gray, are illustrated in Figure 7. Two different scenarios, a relatively strong source-relay correlation (corresponding to small  $p_e$  value) and a relatively weak source-relay correlation (corresponding to large  $p_e$  value) are considered. It can be seen from Figure 7a that with a strong source-relay correlation, the extrinsic information  $I_{e,D_r}^u$  provided by  $D_r$  has a significant effect on  $T_{D_s}^c(\cdot)$ . On the contrary, when the source-relay correlation is weak,  $I_{e,D_r}^u$  has a negligible influence on  $T_{D_s}^c(\cdot)$ , as shown in Figure 7b.

For the proposed DJSCC decoding scheme, both the source memory and the source-relay correlations are exploited in the iterative decoding process. The impact of the source memory and the source-relay correlations on  $D_s$ , represented by the 3D EXIT planes, shown in light-blue, is presented in Figure 7. We can observe that higher extrinsic information can be achieved (EXIT planes are lifted up) by exploiting the source memory and the source-relay correlations simultaneously, which will help decoder  $D_s$  perfectly retrieve the source information sequence even at a low  $\text{SNR}_{sd}$  scenario.

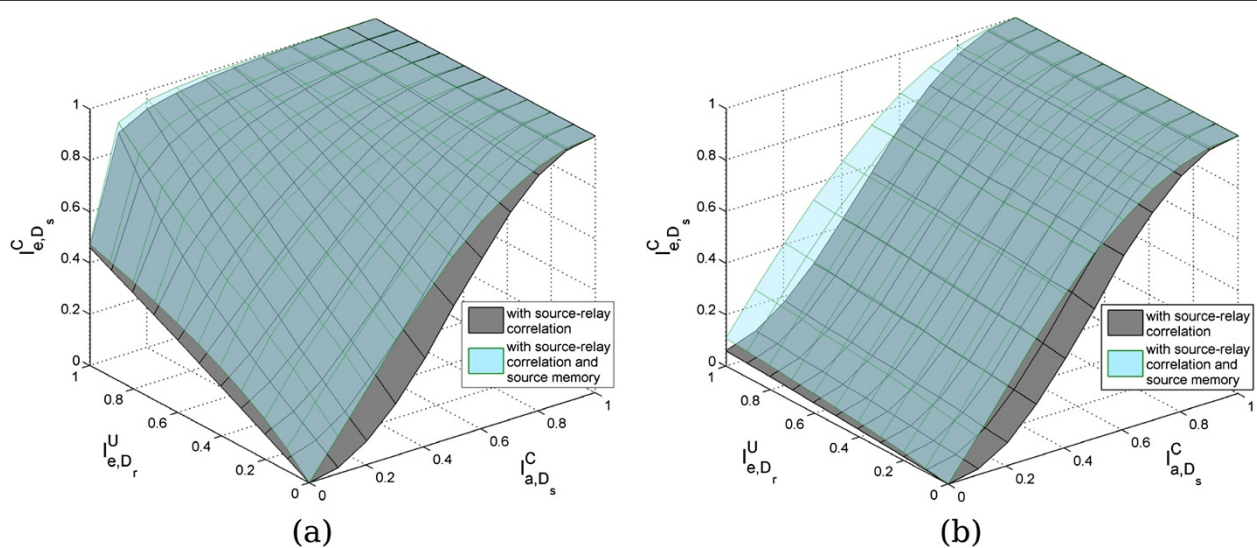
### Convergence analysis and BER performance evaluation

A series of simulations was conducted to evaluate the convergence property, as well as BER performance of the proposed technique. The information sequences are generated from Markov sources with different state transition probabilities. The block length is 10000 bits, and 1000 different blocks were transmitted for the sake of keeping reasonable accuracy. The encoder used at the source and

relay nodes,  $C_s$  and  $C_r$ , respectively, are both memory-1 half rate RSC with generator polynomials  $(G_r, G) = (3, 2)_8$ . Five VIs took place after every HI, with the aim of exchanging extrinsic information to exploit the source-relay correlation. The whole process was repeated 50 times. All the three relay location scenarios were evaluated, with respect to the SNR of the source-destination link. The doping rates are set at  $K_s = K_r = 2$  for location A, while  $K_s = 1$ ,  $K_r = 16$  for both the location B and C. The threshold for estimating  $\hat{p}_e$  [6] is set at 1.

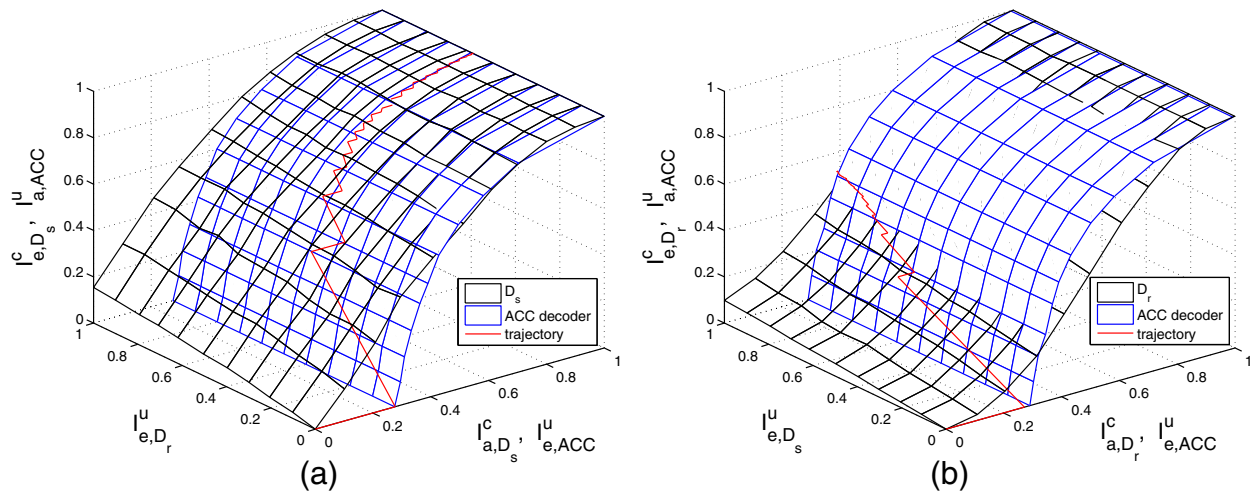
### Convergence behavior with the proposed decoder

The convergence behavior with the proposed DJSCC decoder at the relay location A with  $\text{SNR}_{sd} = -3.5$  dB is illustrated in Figure 8. As described in Section ‘Proposed decoding scheme’, the decoding algorithms for  $D_s$  and  $D_r$  are not the same, and thus the upper and lower HIs are evaluated separately. It can be observed from Figure 8b that the EXIT planes of  $D_r$  and ACC decoder finally intersect with each other at about  $I_{e,D_r}^c = 0.52$ , which corresponds to  $I_{e,D_r}^u = 0.59$ . This observation indicates that  $D_r$  can provide  $D_s$  with  $I_{e,D_r}^u = 0.59$  a priori mutual information via the VI. Figure 8a shows that when  $I_{e,D_r}^u = 0$ , the convergence tunnel is closed, but it is slightly open when  $I_{e,D_r}^u = 0.59$ . Therefore, through extrinsic mutual information exchange between  $D_s$  and  $D_r$ , the trajectory of the upper HI can sneak through the convergence tunnel and finally reach the convergence point while the trajectory of the lower HI gets stuck. It should be noted here that since  $\hat{p}_e$  is estimated and updated during every iteration, the trajectory of the upper HI does not match exactly with the EXIT planes of  $D_s$  and the ACC decoder, especially at the



**Figure 7** The EXIT planes of decoder  $D_s$  with (a)  $p_e = 0.01$  and (b)  $p_e = 0.3$ . The gray planes indicates the case where only the source-relay correlation is exploited, while the light-blue planes indicates the case where both the source-relay correlation and the source memory structure are exploited. For Markov source,  $p_1 = p_2 = 0.8$ ,  $H(S) = 0.72$ .





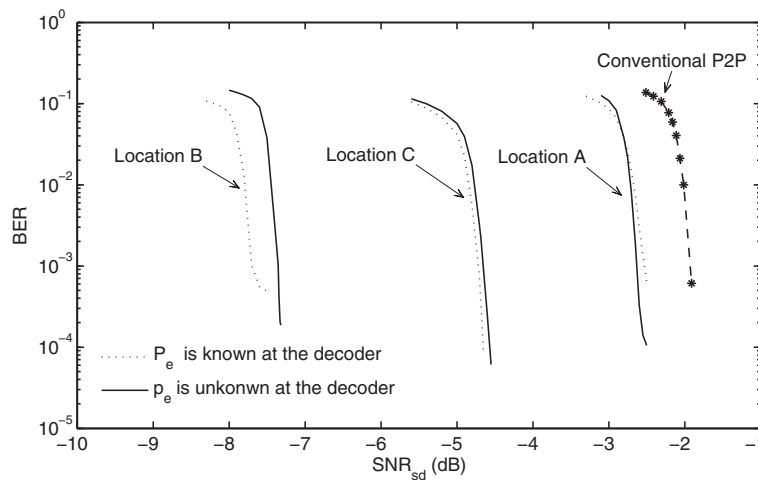
**Figure 8** The 3D EXIT chart analysis for the proposed DJSCC decoder in relay location A,  $\text{SNR}_{sd} = -3.5$  dB. (a) Upper HI with  $f_c$  function, (b) Lower HI with  $f_c$  function.

first several iterations. Similar phenomena is observed for the trajectory of the lower HI.

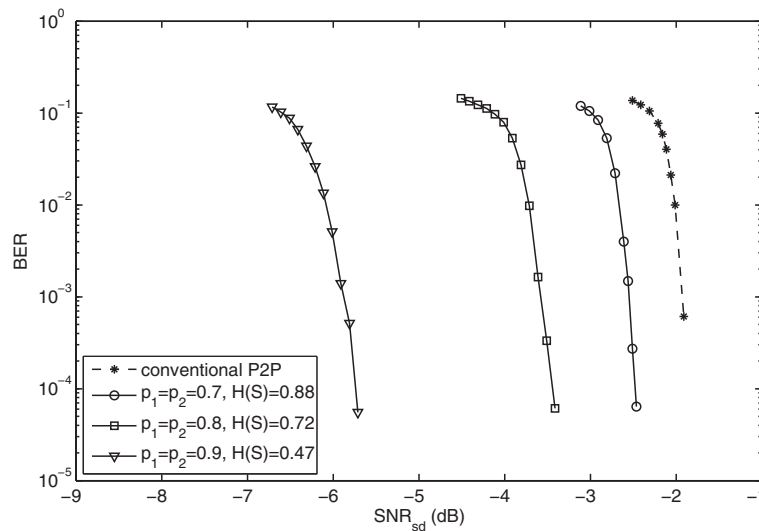
#### Contribution of the source-relay correlation

The performance gains obtained by exploiting only the source-relay correlations largely rely on the quality of the source-relay link (which can be characterized by  $p_e$ ), as described in the previous section. Figure 9 shows the BER performance of the proposed technique when  $p_e$  is known and unknown at the decoder, while the memory structure of Markov source is not taken into account. It can be observed that for relay location A and C, the BER performance of the proposed decoder is almost the same when  $p_e$  is known and unknown at the decoder.

However, for relay location B, convergence threshold is  $-7.7$  and  $-7.4$  dB when  $p_e$  is known and unknown at the decoder, respectively, which results in a performance degradation of  $0.3$  dB. It can also be seen from Figure 9 that, the performance gains obtained by exploiting only source-relay correlation ( $p_e$  is assumed to be unknown at the decoder) for the locations A, B, and C, over the conventional point-to-point (P2P) communication system where *relaying is not involved*, are  $0.6$ ,  $5.4$ , and  $2.6$  dB, respectively. Among these three different relay location scenarios, the quality of the source-relay link with the location A is the worst and that with the location B is the best, if the  $\text{SNR}_{sd}$  is the same. This is consistent with the simulation results.



**Figure 9** The BER performance of the proposed DJSCC decoder for relay systems versus SNR of the source-destination link. Three different relay location scenarios are considered. The memory structure of Markov source is not considered.



**Figure 10** The BER performance of the proposed DJSCC decoder for different Markov sources. Source-relay correlation is not considered, and hence the lower  $H$  is not needed.

### Contribution of the source memory structure

To demonstrate the performance gains obtained by exploiting the memory structure of the Markov source, the BER curves of the proposed DJSCC technique, which only exploits the source memory structure (DJSCC/SM), and hence *relaying is not involved in the scenarios assumed in this section*, are provided in Figure 10, where  $K_s = 1$  was assumed. The BER curve of the conventional P2P communication system that does not exploit the memory structure of source is also provided in the same figure. It can be observed that the performance gain of 0.55, 1.5, and 3.6 dB can be obtained by DJSCC/SM exploiting the memory structure of Markov sources with entropy  $H(S)$  of 0.88, 0.72, and 0.47, respectively. This is consistent with the fact that as the entropy of the source decreases, the performance gain increases.

For the completeness of the article, performance comparison between the DJSCC/SM and the technique proposed in [14], which is referred to as Joint Source Channel Turbo Coding (JSCTC), is provided in this section. The performance gains of DJSCC/SM over conventional P2P system are summarized in Table 1, together with the results of JSCTC as a reference. It can be found from the

table that with the both techniques, substantial gains can be achieved by exploiting the knowledge of the state transition probabilities of the Markov sources. This indicates that exploiting the source memory structure provides us with significant advantage. It should be emphasized that JSCTC uses parallel-concatenated codes and employs two memory-4 constituent codes. On the other hand, our proposed system uses serial-concatenated codes and employs two memory-1 constituent codes. Nevertheless, the proposed DJSCC/SM technique outperforms JSCTC, even though the complexity with our proposed DJSCC/SM technique is much smaller than JSCTC.

### BER performance of the proposed technique

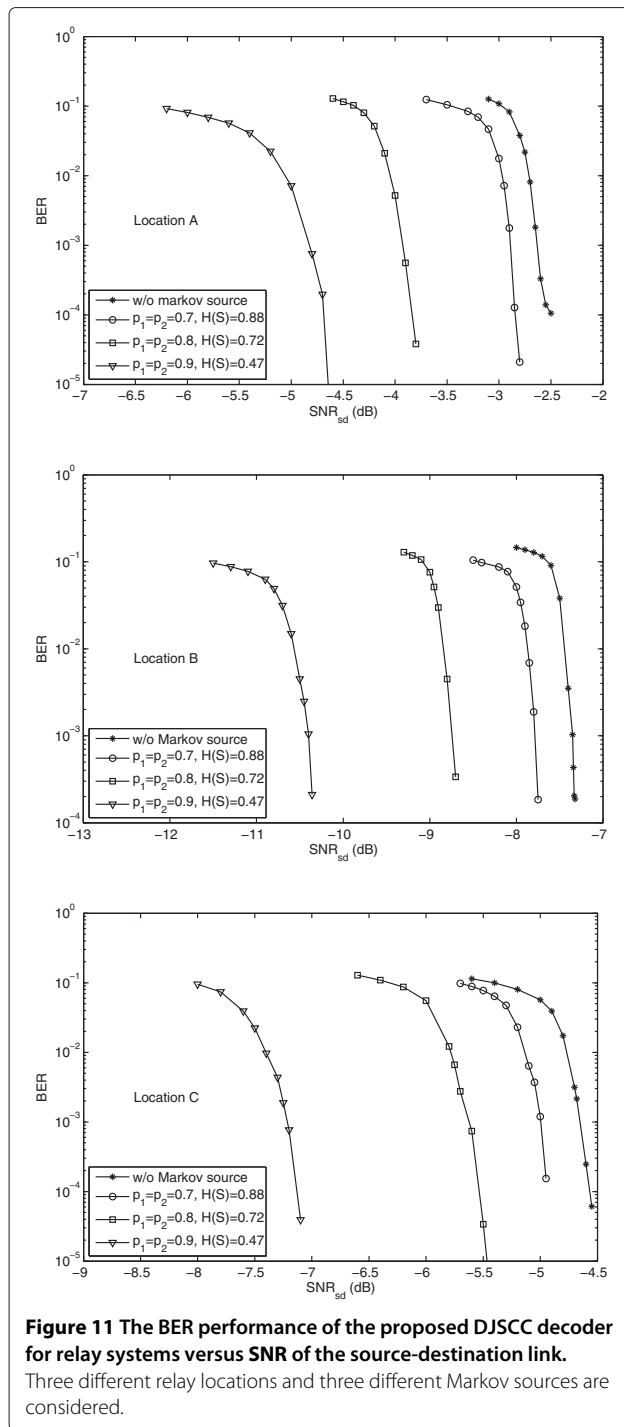
The proposed DJSCC technique exploits both the source-relay correlation and the memory structure of Markov source simultaneously during the iterative decoding process, thus more performance gains should be achieved. The BER performance of the proposed technique for different Markov sources is shown in Figure 11. As a reference, the BER curves of the techniques that only exploit the source-relay correlation are also provided, which are labeled as “w/o Markov source”. The performance gains achieved by the proposed DJSCC technique are summarized in Table 2. It can be observed that by exploiting the memory structure of Markov source, considerable gains can be achieved.

### Application to image transmission

The proposed technique was applied to image transmission to verify the effectiveness of the proposed DJSCC technique. The results with the conventional P2P, the proposed DJSCC technique that only exploit source-relay

**Table 1** BER performance comparison between DJSCC/SM and JSCTC

| Markov source parameters |       |        | Gains over conventional P2P |               |
|--------------------------|-------|--------|-----------------------------|---------------|
| $p_1$                    | $p_2$ | $H(S)$ | JSCTC (dB)                  | DJSCC/SM (dB) |
| 0.7                      | 0.7   | 0.88   | 0.45                        | 0.55          |
| 0.8                      | 0.8   | 0.72   | 1.29                        | 1.5           |
| 0.9                      | 0.9   | 0.47   | 3.03                        | 3.6           |



correlation (DJSCC/SR) and DJSCC/SM are also provided for comparison. Two cases were tested: (A) binary (black and white) image and (B) Grayscale image with 8-digits pixel representations. In (A), each pixel of the image has only two possible values (0 or 1). Binary images are widely used in simple devices, such as laser printers, fax machines, and bilevel computer displays. It is quite

straightforward that the binary image can be modeled as Markov source. A binary image with  $256 \times 256$  pixels and state transition probabilities  $p_1 = 0.9538$  and  $p_2 = 0.9480$  is shown in Figure 12a as an example. The image data is encoded column-by-column. Figures 12b–e show the estimates of the image obtained as the result of decoding at  $\text{SNR}_{sd} = -10$  dB with the conventional P2P technique, DJSCC/SR, DJSCC/SM and DJSCC, respectively. As can be seen from Figure 12, with the conventional P2P transmission, the estimated image quality is the worst containing 43.8% pixel errors (see the figure caption), since neither source-relay correlation nor source memory is exploited. With DJSCC/SR and DJSCC/SM, the estimated images contain 19.4% and 8.1% pixel errors, respectively. The proposed DJSCC that exploits both source-relay correlation and source memory achieves perfect recovery of the image, with 0% pixel error.

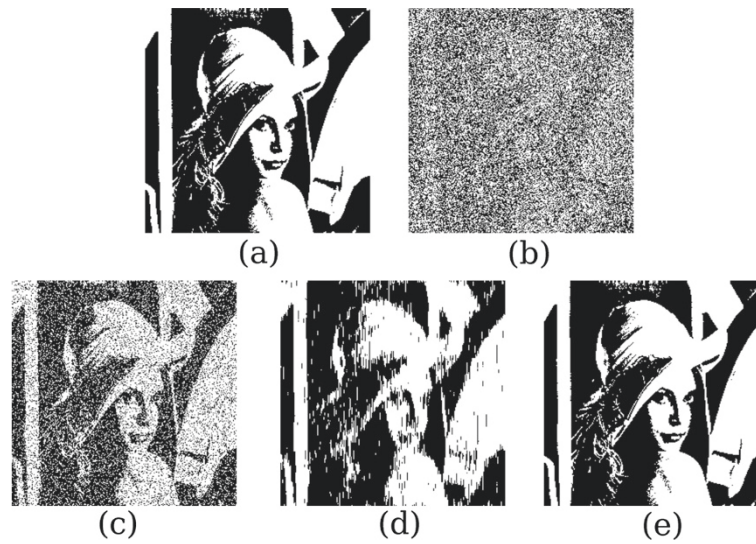
Grayscale images are widely used in some special applications, such as medical imaging, remote sensing and video monitoring. An example of a grayscale image with  $256 \times 256$  pixels is shown in Figure 13a, which is used in the simulation for (B). There are 8 bit planes in this image: the first bit plane contains the set of the most significant bits of each pixel, and the 8th contains the least significant bits, where each bit plane is a binary image. The image data is encoded plane-by-plane and column-by-column within each plane, the average state transition probabilities are  $p_1 = 0.7167$  and  $p_2 = 0.6741$ . Figures 13b–e show the estimates of the image, obtained as the result of decoding, at  $\text{SNR}_{sd} = -7.5$  dB with the conventional P2P, DJSCC/SR, DJSCC/SM, and DJSCC, respectively. It can be observed that the performance with DJSCC/SR (50.27% pixel errors) and DJSCC/SM (96.9% pixel errors) are better than that with conventional P2P (98.1% pixel errors). However, by exploiting source-relay correlation and source memory simultaneously, the proposed DJSCC achieves perfect recovery of the image, with 0% pixel error.

## Conclusion

In this article, we have presented a DJSCC scheme for transmitting binary Markov source in a one-way relay system. The relay does not aim to completely eliminate the errors in the source-relay link. Instead, the relay only

**Table 2** BER performance gains of the DJSCC over the technique that only exploits source-relay correlation

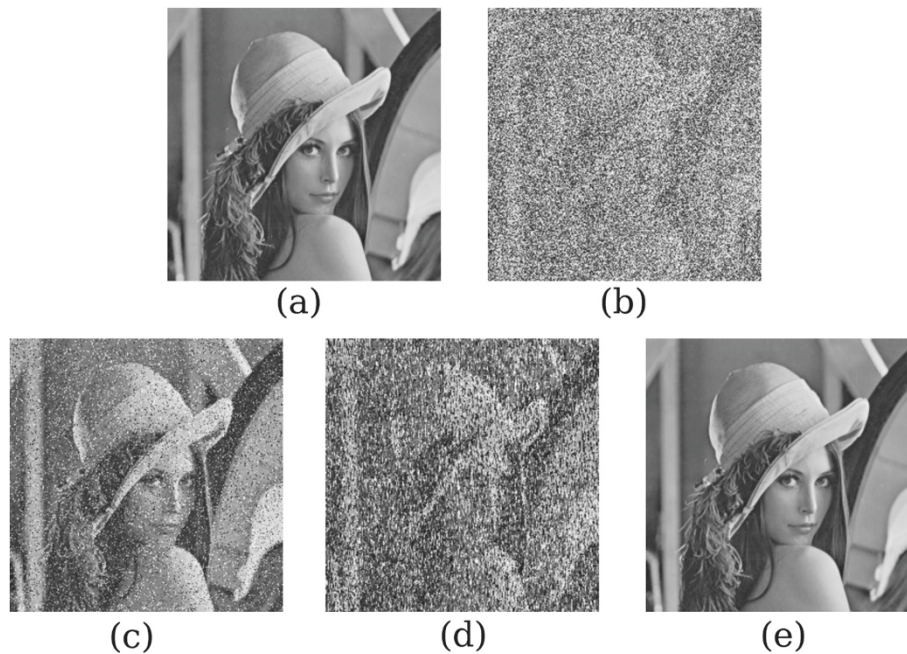
| Markov source parameters |       |        | Relay locations |        |        |
|--------------------------|-------|--------|-----------------|--------|--------|
| $p_1$                    | $p_2$ | $H(S)$ | A (dB)          | B (dB) | C (dB) |
| 0.7                      | 0.7   | 0.88   | 0.3             | 0.45   | 0.45   |
| 0.8                      | 0.8   | 0.72   | 1.2             | 1.4    | 0.9    |
| 0.9                      | 0.9   | 0.47   | 2.2             | 3.05   | 2.6    |



**Figure 12** Image transmission for a binary image with  $p_1 = 0.9538$  and  $p_2 = 0.9480$  at  $\text{SNR}_{sd} = -10$  dB. (a) original transmitted image, (b) conventional P2P (98.1% pixel errors), (c) DJSCC/SR (50.27% pixel errors), (d) DJSCC/SM (96.9% pixel errors), (e) DJSCC (0% pixel errors). The relay location is B.

extracts and forwards the source information sequence to the destination, even though the extracted information sequence may contain some errors. Since the error probability of the source-relay link can be regarded as source-relay correlation, in our proposed technique, the *LLR* updating function is adopted to estimate and exploit

the source-relay correlation. Furthermore, to exploit the memory structure of Markov source, the trellis of Markov source and that of the channel encoder at the source node are combined to construct a super trellis. A modified version of the BCJR algorithm has been derived, based on this super trellis, to perform joint decoding of Markov



**Figure 13** Image transmission for a gray image with  $p_1=0.7167$  and  $p_2=0.6741$  at  $\text{SNR}_{sd}=-7.5$  dB. (a) original transmitted image, (b) conventional P2P (98.1% pixel errors), (c) DJSCC/SR (50.27% pixel errors), (d) DJSCC/SM (96.9% pixel errors), (e) DJSCC (0% pixel errors). The relay location is B.



source and channel code at the destination. By exploiting the source-relay correlation and the memory structure of Markov source simultaneously, the proposed technique can achieve significant gains over the techniques that only exploit the source-relay correlation, which is verified through BER simulations as well as image transmission simulations.

#### Competing interests

The authors declare that they have no competing interests.

#### Acknowledgements

This research was supported in part by the Japan Society for the Promotion of Science (JSPS) Grant under the Scientific Research KIBAN, (B) No. 2360170, (C) No. 2256037, and in part by Academy of Finland SWOCNET project.

#### Author details

<sup>1</sup>School of Information Science, Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan. <sup>2</sup>Centre for Wireless Communications, University of Oulu, P.O. Box 4500, 90014, Oulu, Finland.

Received: 1 March 2012 Accepted: 30 July 2012

Published: 16 August 2012

#### References

1. Z Xiong, AD Liveris, S Cheng, Distributed source coding for sensor networks. *IEEE Signal Process. Mag.* **21**(5), 80–94 (2004)
2. H Li, Q Zhao, Distributed modulation for cooperative wireless communications. *IEEE Signal Process. Mag.* **23**(5), 30–36 (2006)
3. B Zhao, MC Valenti, Distributed turbo coded diversity for relay channel. *Electron Lett.* **39**(10), 786–787 (2003)
4. R Youssef, A Graell i Amat, Distributed serially concatenated codes for multi-source cooperative relay networks. *IEEE Trans. Wirel. Commun.* **10**, 253–263 (2011)
5. Z Si, R Thobaben, M Skoglund, On distributed serially concatenated codes. in *Proc. IEEE 10th Workshop Signal Processing Advances in Wireless Communications SPAWC'09*. (Perugia, Italy, 2009). 653–657
6. J Garcia-Frias, Y Zhao, Near-Shannon/Slepian-Wolf performance for unknown correlated sources over AWGN channels. *IEEE Trans. Commun.* **53**(4), 555–559 (2005)
7. K Anwar, T Matsumoto, Accumulator-assisted distributed Turbo codes for relay system exploiting source-relay correlations. *IEEE Commun. Lett.* **16**(7), 1114–1117 (2012)
8. R Thobaben, J Kliewer, Low-complexity iterative joint source-channel decoding for variable-length encoded Markov sources. *IEEE Trans. Commun.* **53**(12), 2054–2064 (2005)
9. J Kliewer, R Thobaben, Parallel concatenated joint source-channel coding. *Electron. Lett.* **39**(23), 1664–1666 (2003)
10. R Thobaben, J Kliewer, On iterative source-channel decoding for variable-length encoded Markov sources using a bit-level trellis. in *Proc. 4th IEEE Workshop Signal Processing Advances in Wireless Communications SPAWC*. (Auckland, New Zealand, 2003). 50–54
11. M Jeanne, JC Carlsch, P Siohan, Joint source-channel decoding of variable-length codes for convolutional codes and turbo codes. *IEEE Trans. Commun.* **53**, 10–15 (2005)
12. J Garcia-Frias, JD Villasenor, Combining hidden Markov source models and parallel concatenated codes. *IEEE Commun. Lett.* **1**(4), 111–113 (1997)
13. J Garcia-Frias, JD Villasenor, Joint turbo decoding and estimation of hidden Markov sources. *IEEE J. Sel. Areas Commun.* **9**, 1671–1679 (2001)
14. GC Zhu, F Alajaji, Joint source-channel turbo coding for binary Markov sources. *IEEE Trans. Wirel. Commun.* **5**(5), 1065–1075 (2006)
15. K Kobayashi, T Yamazato, H Okada, M Katayama, Joint channel decoding of spatially and temporally correlated data in wireless sensor networks. in *Proc. Int. Symp. Information Theory and Its Applications ISITA*. (Rome, Italy, 2008). 1–5
16. K Anwar, T Matsumoto, Very simple BICM-ID using repetition code and extended mapping with doped accumulator. *Wirel. Personal Commun.* **1**–12 (2011). doi:10.1007/s11277-011-0397-1

17. TM Cover, JA Thomas, *Elements of Information Theory*, 2nd edn. (John Wiley & Sons Inc., New York, 2006)
18. L Bahl, J Cocke, F Jelinek, J Raviv, Optimal decoding of linear codes for minimizing symbol error rate (Corresp.) *IEEE Trans. Inf. Theory.* **20**(2), 284–287 (1974)
19. M Tuchler, Convergence prediction for iterative decoding of threefold concatenated systems, in *Proc. IEEE Global Telecommunications Conf. GLOBECOM'02*. **2**, 1358–1362 (2002)
20. S ten Brink, Code characteristic matching for iterative decoding of serially concatenated codes. *Ann. Telecommun.* **56**, 394–408 (2001)
21. S ten Brink, Convergence behavior of iteratively decoded parallel concatenated codes. *IEEE Trans. Commun.* **49**(10), 1727–1737 (2001)

doi:10.1186/1687-1499-2012-260

**Cite this article as:** Zhou et al.: Distributed joint source-channel coding for relay systems exploiting source-relay correlation and source memory. *EURASIP Journal on Wireless Communications and Networking* 2012 **2012**:260.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](http://springeropen.com)