Partial Soft Decode and Forward

Pir S. Ahmed

Electrical Engineering Department,
Comsats Institute of
Information Technology,
Abbottabad, Pakistan.
Email: shah_gul_khan@ciit.net.pk

Robert G. Maunder and Lajos Hanzo

School of Electronics
and Computer Science,
University of Southampton,
Hampshire, SO17 1BJ, UK.
Email: {rm,lh}@ecs.soton.ac.uk

Abstract—Soft Decode and Forward (SDF) cooperation techniques have been shown to improve the performance of Distributed Turbo Coding (DTC) schemes, in which the Source Node (SN) and Relay Node (RN) employ convolutional codes, while the Destination Node (DN) employs a turbo decoder. However, this technique has not be able to mitigate the typical error floors that are associated with DTC. In this paper, we mitigate these error floors by upgrading the SDF technique to a scheme in which turbo codes are employed at the SN, RN and DN. We demonstrate that this scheme is capable of offering significant gains over both Amplify and Forward (AF) and classic direct-transmission schemes, even if only partial decoding is performed in the RN. More specifically, in our Partial Soft Decode and Forward (P-SDF) scheme, there is no excessive penalty associated with terminating the RN's iterative decoding process before convergence is achieved, as a benefit of the soft nature of its modulation scheme.

I. Introduction

Relay networks [1], [2] have been proposed to facilitate cooperative ad-hoc mobile communications. Here, information is conveyed from the Source Node (SN) to the Destination Node (DN) via a Relay Node (RN). This can be selected from the set of all nodes, allowing every node to serve as source, destination or relay at different times. The careful selection of RNs can provide spatial diversity in fading transmission environments and reduced path losses. As a result, lower transmit powers are facilitated and reduced interference is imposed, yielding significant overall network efficiency gains [3].

Typically, two relaying techniques are used, Decode and Forward (DF) and Amplify and Forward (AF) [4], [5]. In a DF RN, the signal received from the SN is channel decoded and then re-encoded, before it is forwarded to the DN. At the cost of imposing some decoding and encoding complexity upon the RN, this approach can mitigate the effects of noise in the source-relay channel. However, if the RN fails to correctly decode the signal received from the SN, it will propagate errors to the DN [6]. By contrast, an AF RN merely amplifies the signal received from the SN (together with any received noise) and forwards it to the DN. While the amplified noise can induce errors within the DN, the soft nature of the low-

complexity AF approach prevents error propagation from the RN to the DN.

The recently proposed Soft Decode and Forward (SDF) [7] technique resembles DF, except for the difference that decoding and re-encoding are performed in a Soft-Input Soft-Output (SISO) fashion, using the Bahl Cocke Jelinek Raviv (BCJR) algorithm [8], for example. In contrast to the hard information of DF that only identifies *what* the most likely value of each bit is, the soft information of SDF also quantifies *how* likely each value of each bit is. In SDF, the resultant soft information is transmitted to the DN in order to express how much confidence the RN obtained in its decoding decisions. This approach therefore combines the advantages of DF and AF, jointly mitigating both the effects of noise in the source-relay link and the propagation of decoding errors introduced by the RN. Initial efforts [9] achieved SDF by quantizing each soft bit and mapping it to a corresponding sequence of hard bits, which are modulated onto the channel. However, the associated bandwidth expansion can be avoided by adopting the more recent technique of *soft modulation* [6], [7], [10], [11], in which each soft bit is modulated directly onto the channel in an analogue manner.

Recently [6], [7], [10], [11], SDF has been applied in Distributed Turbo Coding (DTC) schemes [12]. Here, the SN broadcasts a convolutional encoded signal to both the RN and the DN. The RN decodes the received signal using a single BCJR operation, interleaves the resultant soft information and then re-encodes it using another BCJR operation [6]. The resultant soft information is transmitted to the DN where the relayed and the directly-received information are decoded using the iterative exchange of soft information between two BCJR decoders.

However, because the RN of a DTC scheme does not employ iterative decoding, it can be more susceptible to decoding errors than the DN. Since, these errors can propagate to the DN, the inferior decoding performance of the RN is likely to dominate the overall performance. This is typically manifested as an error floor in the system's Bit Error Ratio

(BER) performance[1] [6].

This motivates the use of iterative decoding in the RN to mitigate the propagation of decoding errors to the DN, as we propose in this paper. More specifically, we consider the use of a turbo encoder in the transmitter, a soft turbo decoder and encoder in the RN and a turbo decoder in the DN.

However, iterative decoding implies an increased RN complexity. This is undesirable when the RN has scarce energy resources, since these are consumed by forwarding messages that are neither addressed to nor are initiated from the RN. For this reason, we propose a Partial Soft Decode and Forward (P-SDF) technique, which allows the RN to curtail the iterative decoding process before convergence is achieved and convey the soft information obtained using P-SDF. We detail this process in Section II.

In Section III, we use EXtrinsic Information Transfer (EXIT) chart and Bit Error Ratio (BER) analysis to investigate the trade-off between the number of decoding iterations that are performed within the RN and the resultant system performance. These results and the associated RN complexities are compared with those of corresponding AF, DF and soft relaying benchmarkers. We conclude our findings in Section IV.

II. P-SDF SCHEME

Figure 1 shows the block diagram of the proposed P-SDF scheme. The cooperation between the SN and the RN takes
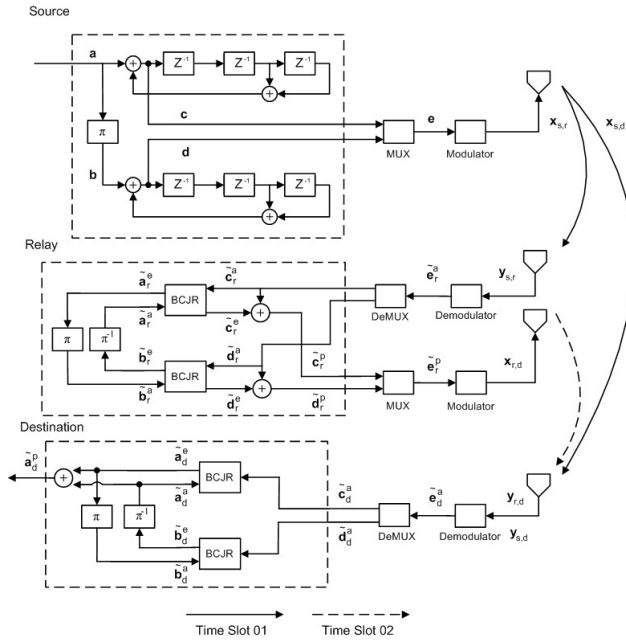

Fig. 1. Schematic of the proposed P-SDF scheme.

[1]This error floor is exacerbated by a further problem that is associated with the application of SDF in DTC schemes. More specifically, the BCJR encoder employed by the RN is required to be non-recursive in DTC schemes [10], otherwise the encoded soft information that it outputs will contain no information. Since the corresponding BCJR decoder employed in the DN is therefore also required to be non-recursive, it will not facilitate iterative decoding convergence toward the maximum likelihood BER performance [13], yielding an increased error floor.

place in a time-division manner. More specifically, the SN encodes the message and transmits it to both the RN and DN during the first time slot. In the second time slot, the RN retransmits the over-heard message to the DN after modifying it according to the P-SDF protocol.

The process commences when the SN encodes the K-bit sequence $\mathbf{a} \in \{0,1\}^K$ using the $R = 0.5$-rate turbo encoder [14] shown in Figure 1. This rearranges the order of the bit sequence $\mathbf{a}$ using the interleaver π, in order to obtain the bit sequence $\mathbf{b}$, as shown in Figure 1. The uncoded bit sequences $\mathbf{a}$ and $\mathbf{b}$ are then encoded using a parallel concatenation of unity-rate recursive convolutional encoders. The resultant bit sequences $\mathbf{c}$ and $\mathbf{d}$ are multiplexed in order to obtain the $N = K/R$-bit sequence $\mathbf{e} \in \{0,1\}^N$, as shown in Figure 1. This is then mapped onto BPSK symbols $\mathbf{x}_s$ and transmitted to both the RN and the DN according to

$$\mathbf{x}_{s,r} = \mathbf{x}_{s,d} = (2\mathbf{e} - 1)\sqrt{E_s}, \tag{1}$$

where E_s is the energy radiated per symbol. Here, subscripts of the form i,j identify the transmitting node i and receiving node j, where s, r and d correspond to the SN, RN and DN, respectively.

The baseband-equivalent transmission over a pathloss channel may be modeled as

$$\mathbf{y}_{i,j} = \frac{|h_{i,j}|}{\sqrt{d_{i,j}^\gamma}}\mathbf{x}_{i,j} + \mathbf{n}_{i,j}. \tag{2}$$

Here, $d_{i,j}$ is the distance between the transmitter and the receiver and γ is the path loss exponent. Furthermore, $|h_{i,j}|$ is the magnitude[2] of the fading coefficient, which is assumed to remain constant for the duration of the transmission. In Additive White Gaussian Noise (AWGN) channels, $h_{i,j} = 1$, while in quasi-static Rayleigh fading channels, the value of $h_{i,j}$ has a complex-valued Gaussian distribution, where the real and imaginary part each have a mean of 0 and a variance of $1/2$. Finally, the elements of the noise vector $\mathbf{n}_{i,j}$ also have a complex-valued Gaussian distribution, where the real and imaginary part each have a mean of 0 and a variance of $\sigma_{i,j}^2$. The noise power spectral density is given by $N_0 = 2\sigma_{i,j}^2$ and the Signal to Noise Ratio (SNR) between the SN and the DN is given by E_s/N_0, when their distance is normalized to $d_{s,d} = 1$.

As shown in Figure 1, the RN's BPSK demodulator outputs soft information in the form of the vector of Logarithmic Likelihood Ratios (LLRs) $\tilde{\mathbf{e}}_r^a$. Here, a particular LLR $\tilde{e}_r^a$ in the vector $\tilde{\mathbf{e}}_r^a$ expresses a soft decision for the corresponding bit e in the vector $\mathbf{e}$ according to [14]

$$\tilde{e}_r^a = \ln\frac{P(e=1|y_{s,r})}{P(e=0|y_{s,r})}, \tag{3}$$

where $y_{s,r}$ is the corresponding received signal. The RN's BPSK demodulator calculates these LLRs according to

$$\tilde{\mathbf{e}}_r^a = \frac{2|h_{s,r}|\sqrt{E_s}}{\sigma_{s,r}^2\sqrt{d_{s,r}^\gamma}}\Re[\mathbf{y}_{s,r}]. \tag{4}$$

[2]We assume that coherent demodulation is employed, which can eliminate the phase rotation imposed by fading.

Here, the RN is assumed to have acquired accurate knowledge of $(2|h_{\mathrm{s,r}}|\sqrt{E_{\mathrm{s}}})/(\sigma_{\mathrm{s,r}}^2\sqrt{d_{\mathrm{s,r}}^\gamma})$ by analyzing the statistics of the received signal $\mathbf{y}_{\mathrm{s,r}}$.

The superscript a in the notation $\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{a}}$ indicates that the corresponding LLRs are employed as *a priori* information in the RN, as indicated by the subscript r. This *a priori* information is demultiplexed to obtain $\tilde{\mathbf{c}}_{\mathrm{r}}^{\mathrm{a}}$ and $\tilde{\mathbf{d}}_{\mathrm{r}}^{\mathrm{a}}$, which are provided to the parallel concatenated BCJR decoders, as shown in Figure 1. These decoders are operated alternately to iteratively produce the LLR sequences $\tilde{\mathbf{a}}_{\mathrm{r}}^{\mathrm{e}}$ and $\tilde{\mathbf{b}}_{\mathrm{r}}^{\mathrm{e}}$, where the superscripts e indicate that these LLR sequences are used as extrinsic information. The order of this extrinsic information is rearranged by the interleaver π or the deinterleaver π^{-1}, in order to provide the *a priori* LLR sequences $\tilde{\mathbf{a}}_{\mathrm{r}}^{\mathrm{a}}$ and $\tilde{\mathbf{b}}_{\mathrm{r}}^{\mathrm{a}}$. As shown in Figure 1, this *a priori* information is employed to assist the operation of the BCJR decoders in the next iteration, allowing them to produce even more reliable extrinsic information. In this iterative manner, the reliability of the extrinsic information gradually increases until the RN's complexity limit is reached or until a particular early-stopping criterion is met, as discussed at the end of this section. In the case where the complexity limit is reached, the BCJR decoders output the extrinsic LLR sequences $\tilde{\mathbf{c}}_{\mathrm{r}}^{\mathrm{e}}$ and $\tilde{\mathbf{d}}_{\mathrm{r}}^{\mathrm{e}}$. These are added to the corresponding *a priori* LLR sequences and multiplexed to obtain the LLR sequence $\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}$, where the superscript p indicates that this LLR sequence is used as *a posteriori* information.

The RN forwards the *a posteriori* LLR sequence $\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}$ to the DN during the second time slot. Here, soft modulation is employed, unless some particular conditions apply, as discussed at the end of this section. Soft modulation is achieved by assuming that the LLRs are Gaussian distributed [15], which is a valid assumption when communicating over an AWGN channel, as we shall show in Section III. More specifically, the LLRs are assumed to fit the model

$$\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}} = \frac{1}{2}\sigma_{\mathrm{r,r}}^2(2\mathbf{e}-1) + \mathbf{n}_{\mathrm{r,r}}, \tag{5}$$

where the elements of $\mathbf{n}_{\mathrm{r,r}}$ have a real-valued Gaussian distribution, with a mean of 0 and a variance of $\sigma_{\mathrm{r,r}}^2$. The RN can estimate $\sigma_{\mathrm{r,r}}$ using the expression

$$\sigma_{\mathrm{r,r}} = J^{-1}[I(\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}})], \tag{6}$$

where the function $J^{-1}(\cdot)$ is defined in the appendix of [16] and the function $I(\cdot)$ is defined immediately below Figure 4 in [17]. Rather than quantizing the LLRs $\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}$ [9], soft BPSK symbols having an average energy of E_{r} are obtained using the expression [6], [7], [10], [11]

$$\mathbf{x}_{\mathrm{r,d}} = \frac{2\sqrt{E_{\mathrm{r}}}}{\sigma_{\mathrm{r,r}}\sqrt{\sigma_{\mathrm{r,r}}^2+4}}\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}. \tag{7}$$

The transmission of these symbols to the DN may be modeled using Equation 2.

Note that the elements of both $\mathbf{n}_{\mathrm{r,d}}$ in Equation 2 and $\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}$ in Equation 7 are Gaussian distributed. As a result, the signal $\mathbf{y}_{\mathrm{r,d}}$ received at the DN appears to be a conventional BPSK

signal that has been corrupted by AWGN. More specifically, the signal appears to have a transmit energy per symbol of

$$\hat{E}_{\mathrm{r}} = \frac{\sigma_{\mathrm{r,r}}^2 E_{\mathrm{r}}}{\sigma_{\mathrm{r,r}}^2+4}, \tag{8}$$

while the real part of the noise appears to have the variance

$$\hat{\sigma}_{\mathrm{r,d}}^2 = \frac{4E_{\mathrm{r}}}{d_{\mathrm{r,d}}^\gamma(\sigma_{\mathrm{r,r}}^2+4)} + \sigma_{\mathrm{r,d}}^2. \tag{9}$$

For this reason, the soft-modulated signal provided by the RN and the hard-modulated signal provided by the SN can be demodulated in the same manner, before the results are combined according to

$$\tilde{\mathbf{e}}_{\mathrm{d}}^{\mathrm{a}} = \frac{2|h_{\mathrm{s,d}}|\sqrt{E_{\mathrm{s}}}}{\sigma_{\mathrm{s,d}}^2\sqrt{d_{\mathrm{s,d}}^\gamma}}\Re[\mathbf{y}_{\mathrm{s,d}}] + \frac{2|h_{\mathrm{r,d}}|\sqrt{\hat{E}_{\mathrm{r}}}}{\hat{\sigma}_{\mathrm{r,d}}^2\sqrt{d_{\mathrm{r,d}}^\gamma}}\Re[\mathbf{y}_{\mathrm{r,d}}]. \tag{10}$$

Here, the DN is assumed to have acquired accurate knowledge of $(2|h_{\mathrm{s,d}}|\sqrt{E_{\mathrm{s}}})/(\sigma_{\mathrm{s,d}}^2\sqrt{d_{\mathrm{s,d}}^\gamma})$ and $(2|h_{\mathrm{r,d}}|\sqrt{\hat{E}_{\mathrm{r}}})/(\hat{\sigma}_{\mathrm{r,d}}^2\sqrt{d_{\mathrm{r,d}}^\gamma})$ by analyzing the statistics of the received signals $\mathbf{y}_{\mathrm{s,d}}$ and $\mathbf{y}_{\mathrm{r,d}}$, respectively.

Finally, the LLR sequence $\tilde{\mathbf{e}}_{\mathrm{d}}^{\mathrm{a}}$ is employed as *a priori* information for the DN's turbo decoder. This employs the same iterative exchange of extrinsic information as the turbo decoder employed in the RN. However, in the DN, iterations are continued until convergence is achieved, whereupon no further information can be gleaned by performing additional iterations. When this occurs, the *a posteriori* LLR sequence $\tilde{\mathbf{a}}_{\mathrm{d}}^{\mathrm{p}}$ is obtained as shown in Figure 1 and a hard decision is used to reconstruct each bit in the sequence $\mathbf{a}$.

Note that the K-bit source sequence $\mathbf{a}$ is assumed to include a perfect Cyclic Redundancy Check (CRC), which may trigger the early stopping of the RN's iterative decoding process. After each BCJR decoding operation performed during this process, the RN can make a hard decision on the basis of the *a priori* and extrinsic LLR sequences, according to $\hat{\mathbf{a}}_{\mathrm{r}}^{\mathrm{p}} = (\mathrm{sgn}(\tilde{\mathbf{a}}_{\mathrm{r}}^{\mathrm{a}} + \tilde{\mathbf{a}}_{\mathrm{r}}^{\mathrm{e}}) + 1)/2$, respectively. If the resultant bit sequence $\hat{\mathbf{a}}_{\mathrm{r}}^{\mathrm{p}}$ satisfies the perfect CRC, then the RN has obtained $\mathbf{a}$. When this occurs, the RN uses the same encoding process as the SN to obtain the bit sequence $\hat{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}$ and performs hard modulation in analogy with (1), rather than using the soft modulation of (7). In this case, the DN performs demodulation using (10) as usual, but with $\hat{E}_{\mathrm{r}} = E_{\mathrm{r}}$ and $\hat{\sigma}_{\mathrm{r,d}}^2 = \sigma_{\mathrm{r,d}}^2$. Hard modulation is also employed if the early-stopping criterion was not satisfied, but $I(\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}) > 0.85$, which was found experimentally to be the highest value for which soft modulation out-performs hard modulation. At higher values of $I(\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}})$, the LLRs have a high variance $\sigma_{\mathrm{r,r}}^2$, causing soft modulation to convey some LLRs using an excessively high energy, at the cost of conveying some with an excessively low energy. Hard modulation overcomes this problem by using a hard decision to obtain the bit sequence $\hat{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}$ according to $\hat{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}} = (\mathrm{sgn}(\tilde{\mathbf{e}}_{\mathrm{r}}^{\mathrm{p}}) + 1)/2$ and modulating each bit with the same energy, in analogy with (1).

III. SIMULATION RESULTS

The operation of the proposed P-SDF scheme was simulated in order to characterize and compare its performance with that of the corresponding benchmarkers. We consider the scenario where the SN conveys $K = 10000$-bit sequences $\mathbf{a}$ over AWGN channels having a free-space path-loss exponent of $\gamma = 2$. We assume that the transmission is conveyed to a DN at a distance of $d_{s,d} = 1$ and a RN at a distance of $d_{s,r} = 0.5$. Using P-SDF, the RN forwards the bit sequences using a transmit energy per symbol of $E_r = E_s$ to the DN, which is situated at a distance of $d_{r,d} = 0.5$. We assume that each bit sequence $\mathbf{a}$ is interleaved using a different pseudo-random interleaver π.

Figure 2 provides EXIT functions and iterative decoding trajectories for the RN's turbo decoder. As may be expected,

Fig. 3. Distribution of the RN's *a posteriori* LLRs $\tilde{\mathbf{e}}_r^p$ following the completion of 1, 2, 3 and 4 decoding iterations, at an E_b/N_0 value of -2 dB.

and theoretical results presented in Figure 3 reveals that the associated assumption is valid.

Figure 4 provides the DN's EXIT chart at an E_b/N_0 value of -1 dB, following the completion of various numbers of decoding iterations in the RN. As may be expected, this
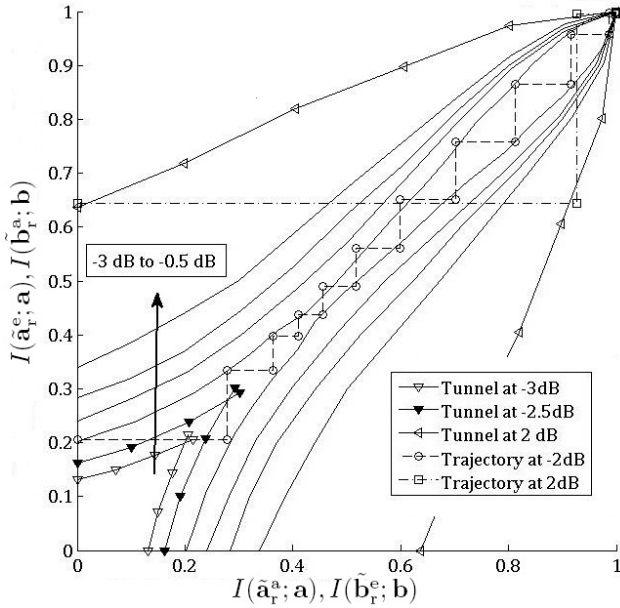

Fig. 2. RN's EXIT chart and decoding trajectories for various E_b/N_0 values.

this EXIT chart resembles that of a conventional turbo decoder. Here, an open EXIT chart tunnel can be created at $E_b/N_0 = 1/R \cdot (E_s + E_r)/N_0$ values that are above a threshold of approximately -2 dB. At this E_b/N_0 value, 11 decoding iterations are typically required to reach the $(1, 1)$ point, where a low BER is achieved [18]. By contrast, Figure 2 shows that just two iterations are required when $E_b/N_0 = 2$ dB.

In Figure 3 we examine the LLRs in the RN's *a posteriori* sequence $\tilde{\mathbf{e}}_r^p$, following the completion of 1, 2, 3 and 4 decoding iterations. In each case, the distribution of the LLRs corresponding to the zero-valued bits in $\mathbf{e}$ is provided separately to that corresponding to one-valued bits. Furthermore, (6) was employed to calculate the $\sigma_{r,r}$ value for the *a posteriori* LLR sequence $\tilde{\mathbf{e}}_r^p$ obtained after each number of decoder iterations. These $\sigma_{r,r}$ values were then employed to obtain the theoretical LLR distributions that are assumed by (5), as shown in Figure 3. The strong match between the simulated
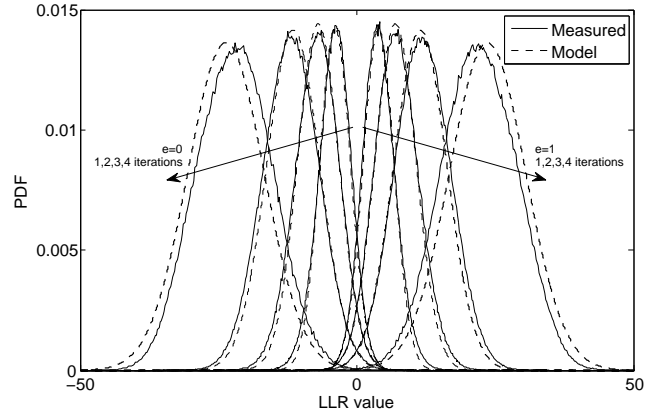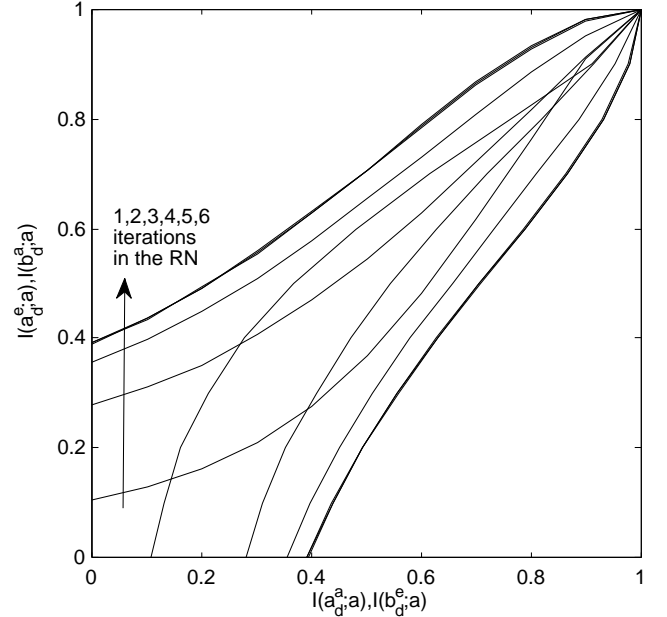

Fig. 4. DN's EXIT chart at an E_b/N_0 value of -1 dB, following the completion of various numbers of decoding iterations in the RN.

reveals that the DN's EXIT chart tunnel widens as more decoding iterations are performed in the RN. The tunnel can be seen to be closed when the RN performs only one iterations, marginally open when two iterations are performed and fully open following the completion of three or more iterations.

The BER performance of the proposed P-SDF scheme is evaluated in Figure 5, for a range of E_b/N_0 values and for the cases where 1, 2, 3, 4, 5 and 6 decoding iterations are performed in the RN. Figure 5 also characterizes the BER performance of various benchmarkers. Like the proposed P-
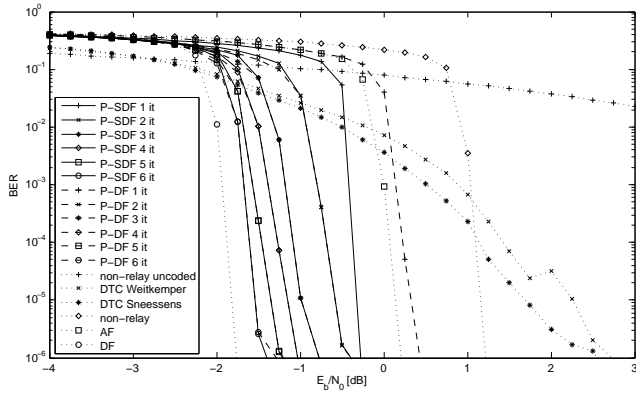
Fig. 5. BER performance of the proposed P-SDF scheme, in addition to that of various benchmarkers.

SDF scheme, the P-DF benchmarker performs only a limited number of decoding iterations in the RN. However, instead of employing the soft modulation of (7), the P-DF scheme always employs the hard modulation technique that the P-SDF scheme adopts when the early stopping criteria is met, as described at the end of Section II. This technique is also adopted by the DF benchmarker, but using as many decoding iterations as is necessary to achieve convergence in the RN. No decoding iterations are performed in the RN of the AF scheme. Instead, $\mathbf{x}_{s,d}$ is obtained by simply normalizing $\mathbf{y}_{s,r}$ so that its average energy per symbol is E_r. In addition to these relay-aided benchmarkers, Figure 5 also characterizes the BER performance of classic direct-transmission turbo-coded and uncoded schemes, which are referred to as the 'non-relay' and the 'non-relay uncoded' schemes, respectively. Finally, Figure 5 plots the BER performance of the SDF DTC schemes of [7] and [10], which are referred to as the 'DTC Sneessens' and 'DTC Weitkemper' schemes, respectively. Note that $E_b/N_0 = 1/R \cdot (E_s + E_r)/N_0$ for the P-SDF, P-DF, DF and AF schemes, $E_b/N_0 = 1/R \cdot E_s/N_0$ for the non-relay schemes and $E_b/N_0 = (2E_s + E_r)/N_0$ for the DTC schemes.

As shown in Figure 5, the proposed P-SDF scheme performs at least as well as the P-DF benchmarker, regardless of how many decoding iterations are performed in the RN. Indeed, when only one decoding iteration is performed, the P-SDF scheme offers a 0.6 dB gain over the P-DF scheme. Furthermore, when a higher complexity limit is affordable in the RN, the performance of the P-SDF scheme can be seen to approach that of the DF scheme, which performs as many decoding iterations in the RN as is required to reach convergence. Finally, the P-SDF scheme offers significant gains over the AF and classic direct-transmission benchmarkers, regardless of how many decoding iterations are performed in the RN. In particular, the proposed P-SDF scheme can be seen to offer significant gains over the DTC benchmarkers, owing to their elevated error floors, as discussed in Section I.

IV. CONCLUSIONS

Previously the SDF cooperation technique has been applied in DTC schemes, where the SN and the RN employ convolutional codes, while the DN employs a turbo decoder. In this paper, we have adapted the SDF technique to a scheme in which turbo codes are employed at the SN, RN and DN. This has the advantage of avoiding the error floors that are associated with the application of SDF in DTC schemes. Furthermore, we have demonstrated a novel P-SDF scheme can offer significant gains over AF and classic direct-transmission schemes, without continuing the RN's iteration decoding process until convergence is achieved, owing to the soft nature of its modulation scheme.

REFERENCES

[1] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity - Part I: System description," *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1927–1938, Nov. 2003.

[2] ——, "User cooperation diversity - Part II: Implementation aspects and performance analysis," *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1939–1948, Nov. 2003.

[3] P. Liu, Z. Tao, Z. Lin, E. Erkip, and S. Panwar, "Cooperative wireless communications: A cross-layer approach," *IEEE Wireless Commun.*, pp. 84–92, Aug. 2006.

[4] J. N. Laneman and G. W. Wornell, "Energy-efficient antenna sharing and relaying for wireless networks," in *Proc. IEEE Wireless Commun. Networking Conf.*, Orlando, FL, USA, Mar. 2002, pp. 7–12.

[5] J. N. Laneman, G. W. Wornell, and D. N. C. Tse, "An efficient protocol for realizing cooperative diversity in wireless networks," in *Proc. IEEE Int. Symp. Inf. Theory*, Washington, DC, USA, June 2002, p. 294.

[6] Y. Li, B. Vucetic, T. Wong, and M. Dohler, "Distributed turbo coding with soft information relaying in multihop relay networks," *IEEE J. Select. Areas Commun.*, vol. 24, no. 11, pp. 2040–2050, Nov. 2006.

[7] H. H. Sneessens and L. Vandendorpe, "Soft decode and forward improves cooperative communications," in *Proc. IEE Int. Conf. on 3G and Beyond*, London, UK, Nov. 2005, pp. 73–76.

[8] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (Corresp.)," *IEEE Trans. Inform. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[9] Y. Hairej, A. Darmawan, and H. Morikawa, "Cooperative Diversity using Soft Decision and Distributed Decoding," in *Proc. IST Mobile and Wireless Communications Summit*, Budapest, Hungary, July 2007.

[10] P. Weitkemper, D. Wubben, V. Kuhn, and K.-D. Kammeyer, "Soft information relaying for wireless networks with error-prone source-relay link system model and basics of distributed coding schemes," in *Proc. Int. ITG Conf. on Source and Channel Coding*, Ulm, Germany, Jan. 2008.

[11] Y. Qi, R. Hoshyar, and R. Tafazolli, "Performance evaluation of soft decode-and-forward in fading relay channels," in *Proc. IEEE Veh. Technol. Conf.*, Singapore, May 2008, pp. 1286–1290.

[12] B. Zhao and M. C. Valenti, "Distributed turbo coded diversity for relay channel," *Electron. Lett.*, vol. 39, no. May, pp. 36–37, 2003.

[13] J. Kliewer, A. Huebner, and D. J. Costello, "On the achievable extrinsic information of inner decoders in serial concatenation," in *Proc. IEEE Int. Symp. Inform. Theory*, Seattle, WA, USA, July 2006, pp. 2680–2684.

[14] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes (1)," in *Proc. IEEE Int. Conf. on Communications*, vol. 2, Geneva, Switzerland, May 1993, pp. 1064–1070.

[15] D. Divsalar, S. Dolinar, and F. Pollara, "Iterative turbo decoder analysis based on density evolution," *IEEE J. Select. Areas Commun.*, vol. 19, no. 5, pp. 891–907, May 2001.

[16] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 670–678, Apr. 2004.

[17] J. Hagenauer, "The EXIT chart - Introduction to extrinsic information transfer in iterative processing," in *Proc. European Signal Processing Conf.*, Vienna, Austria, Sept. 2004, pp. 1541–1548.

[18] S. ten Brink, "Convergence of iterative decoding," *Electron. Lett.*, vol. 35, no. 10, pp. 806–808, May 1999.