Distributed Decoding in Cooperative Communications

Marjan Karkooti and Joseph R. Cavallaro
Rice University,
Department of Electrical and Computer Engineering,
Houston, TX, 77005
{marjan,cavallar}@rice.edu

Abstract— **In this paper, we present a novel relaying strategy called distributed and partial decoding. This strategy can be viewed as a variation of the decode and forward with the difference that the relay partially decodes the signal, re-transmits it to the destination, and the destination continues the decoding. By distributing the decoding process between the relay and the destination, the relay uses less processing power and less time. This is very suitable for practical applications in which relays are battery-operated (such as handsets) and do not want to use all their battery power on relaying the data of other users.**

I. Introduction

Cooperative communication has attracted major attention in the field of wireless communications in recent years because each day more wireless devices enter the market. Until now most of these devices have been working as a stand-alone node communicating with the base station. Most of these devices are in idle mode for long periods of time and are only used for a short time. This results in under-utilized radios and inefficient use of the network resources.

The demand for future generations of wireless devices with higher data rates, longer battery life and lower cost may result in a shift of paradigm in these systems. Multiple antenna systems have shown to be very promising in increasing the data rates. But, most of the mobile devices have limitations in terms of size and battery life that prevent them from supporting MIMO.

Cooperative communication enables these single antenna devices to benefit from transmit diversity by grouping the nodes in vicinity. These single antenna nodes form a virtual antenna array and help each other in transmission. Many papers have been published in recent years analyzing the benefits of cooperative communications and relay channels [1], [2].

In this paper, we propose a new cooperation scheme called distributed and partial decoding. This is a variation of the decode and forward strategy, but with the benefits of reduced processing latency, reduced power consumption and smaller hardware complexity at the relay. This scheme can be used with any error correcting code with iterative decoding, such as low density parity check (LDPC) codes or turbo codes. In this paper we present the results for LDPC codes but the same approach can be applied to other codes too.

The paper is organized as follows. Section II will give an overview of cooperative communications. Our system model is described in section III. Section IV describes LDPC codes and their decoding. The proposed distributed and partial decoding will be discussed in section V. Simulation results and discussion will follow in sections VI and VII. Section VIII will conclude the paper.

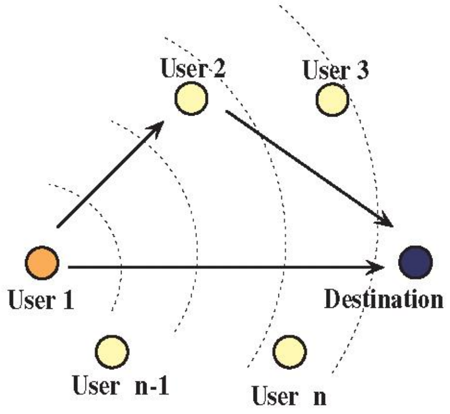


Fig. 1. Demonstration of cooperative wireless transmission.

II. Cooperative Communications

The terms 'cooperative communications' or 'relay channels' refer to a communication system in which neighboring nodes help each other in the transmission. This way, one of the users that does not have any information to send, acts as a relay and helps to transmit the data of other users. In Figure 1, $user1$ or *source* is a wireless device that needs to send data to the destination. When it broadcasts its data, $user2$ hears the signal as well as the *destination.* Since the distance between $user2$ and $user1$ is less than between $user1$ and the destination, $user2$ receives a stronger signal and acts as a relay, either processing the signal before re-transmission or just retransmitting the data to the destination. Another scenario is when the channel between the source and destination is blocked or goes into deep fading. The destination receives two copies of the same signal and can retrieve the input signal with less errors by taking advantage of the transmit diversity. This results in a more robust and more reliable communication.

Several strategies for user cooperation have been proposed in the literature depending on the type of processing at

978-1-4244-2110-7/08/$25.00 ©2007 IEEE

the relay including: *Amplify and Forward* [3], *Decode and Forward* [4], *Estimate-and-Forward* [2], *Coded Cooperation* [5], *Dynamic Relaying or hybrid* [3], *Incremental Relaying*, etc.

In this work, we focus on the decode and forward strategy. We refer the reader to the literature for the discussion about other relaying schemes. In decode and forward strategy, the source encodes the signal with any channel coding scheme such as low density parity check (LDPC) codes, turbo codes, etc. The relay decodes the signal and re-encodes it with either the same code or a new code. Then, it transmits the new codeword to the destination. The destination uses both copies of the signal received from the source and the relay; combines and decodes them.

III. System Model

Here, we present the parameters of our system. In this paper, we assume a three-node communication link between the source, the relay and the destination. Although, our results can be generalized to two or more relays in the network, to simplify the discussion, we will focus on one relay node. The relay is assumed to be half-duplex, which is either receiving or transmitting the data at each time instance. This restriction comes from practice, in which it is extremely difficult and expensive to build a relay that can work in full-duplex mode. The problem comes from the fact that the signal that is being transmitted is generally about $100dB$ stronger than the received signal and the relay has to have proper shielding to eliminate the effects of the transmitting signal.

We assume that the system works in two consecutive time slots t and $t' = 1 - t$. In first time slot t, the source works in the broadcast (BC) mode and transmits the data to both the relay and the destination. In the second time slot t', the source and the relay work in multiple access (MAC) mode. During the MAC mode, the destination receives two signals, one from the source and the other one from the relay.

The channel between the nodes is an additive white gaussian noise (AWGN) channel and the modulation is binary phase shift keying (BPSK). Our results hold for slow fading channels that are constant during the transmission of one block. To be fair in our comparisons with the direct link, we assume that the total transmission power of the system is fixed. The total transmission power for the source in direct transmission is equal to the total power used by the source and the relay during cooperation. Therefore, we assume a global power constraint as follows:

$$tP_{S_{BC}} + t'(P_{S_{MAC}} + P_{R_{MAC}}) \leq P. \qquad (1)$$

in which $P_{S_{BC}}$ is the source transmission power in the BC mode, $P_{S_{MAC}}$ and $P_{R_{MAC}}$ are the source and relay transmission powers in the MAC mode, and P is the total system transmission power. This assumption does not limit our results and in cases where the relay can add extra power to the system, using the relay will be more beneficial.

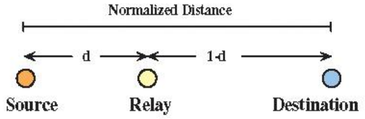
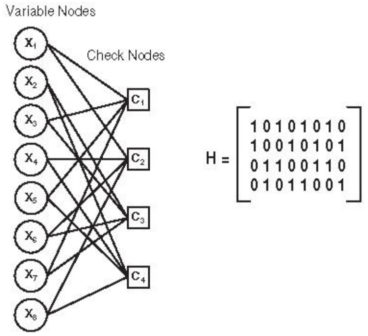


Fig. 2. System model.



$$H = \begin{bmatrix} 1&0&1&0&1&0&1&0 \\ 1&0&0&1&0&1&0&1 \\ 0&1&1&0&0&1&1&0 \\ 0&1&0&1&1&0&0&1 \end{bmatrix}$$

Fig. 3. Tanner graph and the parity check matrix corresponding to a LDPC code.

We assume that the relay lies on a direct path between the source and the destination, with normalized distance of $0 < d < 1$ between the source and the relay (See Figure 2). The channel gain for the SD, SR and RD links are given by:

$$\gamma_{SD} = 1, \gamma_{SR} = \frac{1}{d^{\alpha}}, \gamma_{RD} = \frac{1}{(1-d)^{\alpha}} \qquad (2)$$

in which α is the path-loss exponent [6] and typically ranges from 2 (e.g. free-space propagation) to 4 (e.g. typical rural areas). We assume $\alpha = 2$ in this paper, larger values of α increase the attenuation of the signal by the channel which makes relays and our results more favorable.

Since we are considering the decode and forward relaying strategy, we need to use a channel coding scheme as part of our discussion. Low density parity check codes are one of the best options. The next section will give an overview of these codes and their decoding process.

IV. Low Density Parity Check Codes

Low Density Parity Check code is a linear block code with very good performance close to the Shannon limit. LDPC codes are represented by a very sparse parity check matrix (PCM) $H_{(N-K)\times N}$. The PCM consists of zeros and ones which show the connections between different processing nodes in the bi-partite Tanner graph (See Fig. 3). In this graph, a variable node corresponds to a 'coded bit' or a PCM column, and a check node corresponds to a parity check equation or a PCM row. There is an edge between each pair of nodes if there is a 'one' in the corresponding PCM entry. During the decoding, reliability messages are passed through the graph edges.

In theory, random LDPC codes of infinite block length (10^7 bits or more) perform very close to the channel capacity. Although these codes have very good performance, in practice, it is very difficult or even impossible to design hardware that supports these random codes with very large block lengths. Current technologies impose restrictions on the block length and structure of these codes. Block lengths of $10^3 \sim 10^4$ bits are feasible. Also, cyclic-shifted block structured codes enable efficient hardware design with minor performance loss compared to the random codes.

The LDPC codes are usually decoded using an iterative decoding algorithm called sum-product or belief propagation algorithm. In each iteration some of the errors in the codeword are corrected until all the parity check equations satisfy and the decoded codeword converges to a correct codeword. Irregular LDPC codes usually show better performance than regular codes because the nodes with higher degrees converge faster and assist the nodes with lower degrees.

The LDPC decoder used in this paper utilizes the iterative layered belief propagation (LBP) algorithm as defined in [7]. Here, we present an overview of this decoding algorithm.

A. LDPC Decoding–Layered Belief Propagation Algorithm

The layered belief propagation algorithm is a variation of the standard belief propagation [8], and achieves about two times faster decoding convergence due to the optimized scheduling of reliability messages [9]. The PCM can be viewed as a group of concatenated horizontal layers in which every layer represents the component code. The belief propagation algorithm is repeated for each horizontal layer and updated *a posteriori* probability ratio(APP) messages are passed between layers. Let R_{mj} denote the check node LLR message sent from the check node m to the variable node j. Let $L(q_{mj})$ denote the variable node LLR message sent from the variable node j to the check node m. The $L(q_j)$ $(j = 1, \dots, N)$ represent the APP messages for all variable nodes (coded bits). The APP messages are initialized with the corresponding *a priori* (channel) reliability value of the coded bits. For each variable node j inside the current horizontal layer, messages $L(q_{mj})$ that correspond to a particular check equation m are computed according to:

$$L(q_{mj}) = L(q_j) - R_{mj}. \tag{3}$$

For each check node m, messages R_{mj}, corresponding to all variable nodes j that participate in a particular parity-check equation, are computed according to:

$$R_{mj} = \prod_{j\prime \in N(m) \backslash \{j\}} \operatorname{sign}\left(L(q_{mj\prime})\right) \Psi \left[\sum_{j\prime \in N(m) \backslash \{j\}} \Psi\left(L(q_{mj\prime})\right) \right], \tag{4}$$

where $N(m)$ is the set of all variable nodes from parity-check equation m, and $\Psi(x) = -\log\left[\tanh\left(\frac{|x|}{2}\right)\right]$. For the purpose of more efficient architecture implementation, updating of the check node messages in (4) is replaced with the modified min-sum approximation [10]. According to this solution, the updating of check node messages in the mth row of the kth decoding iteration is determined as:

$$R_{mj} \approx \prod_{j\prime \in N(m) \backslash \{j\}} \operatorname{sign}\left(L(q_{mj\prime})\right) \times \max\left(\min_{j\prime \in N(m) \backslash \{j\}} |L(q_{mj\prime})| - \beta, 0 \right), \tag{5}$$

where β is a correcting offset equal to a positive constant. The *a posteriori* reliability messages in the current horizontal layer are updated according to:

$$L(q_j) = L(q_{mj}) + R_{mj}. \tag{6}$$

Hard decisions can be made after every horizontal layer based on the sign of $L(q_j)$, $j = 1, \dots, n$. If all parity-check equations are satisfied or the pre-determined maximum number of iterations is reached, then the decoding algorithm stops. Otherwise, the algorithm repeats from Eq. (3) for the next horizontal layer.

V. Distributed and Partial Decoding

In [11] we analyzed two schemes of decode, re-encode and forward (DEF) and decode, no encode and forward (DNEF) and showed the benefits of DNEF over DEF. Here we present a brief overview of each of these two schemes and refer the reader to [11] for more detailed discussion.

A. Relay: Decode, No Encode and Forward (DNEF)

In DNEF, we assume that in the broadcast mode, the source encodes the information and sends it to both R and D. The relay receives the codeword, fully decodes it, and retransmits the resulting codeword to the destination in multiple access (MAC) mode. Note that the relay does not re-encode the signal, it keeps the parity bits from the original codeword. These bits have been corrected along with the information bits during the decoding process. The relay transmits the decoded and corrected codeword directly to the destination. This idea has been proposed in [12] which shows that there is benefit in terms of the computations in the relay and power consumption. One advantage of this scheme is that the relay does not need to have specialized hardware for the encoding, and spends less time and power processing other users' data.

We assume that the source is silent in MAC mode and the destination only receives one copy of the signal in the second time frame. We also assume that the two time intervals for BC and MAC modes are equal ($t = t' = 1/2$). The destination receives two different copies of the signal from S and R and performs Maximal Ratio Combining (MRC) on the two signals.

B. Relay: Decode, Re-encode and Forward (DEF)

This case is the original decode and forward strategy in which the relay fully decodes the received signal, re-encodes and re-transmits it to the destination. We assume that the relay uses either the same parity check matrix as the source for re-encoding or a new parity check matrix. The former case has some advantages in situations that there are still errors in the parity bits after the relay decodes the codeword received from the source. The old parity bits are discarded and a new codeword is calculated at the relay. The latter case can be useful when the transmission time for RD is less than the time for SD, so a different block length/ code rate should be used to adjust the block length.

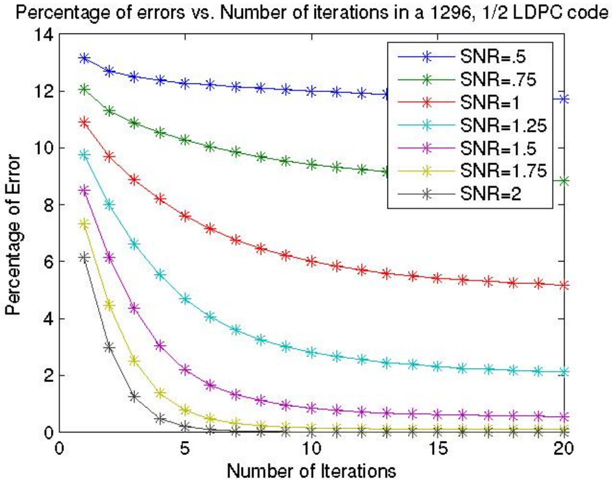


Fig. 4. Percentage of bits in error vs. Number of decoding iterations for different SNR values. For a LDPC code of (1296 ,1/2). Results are for the direct link case

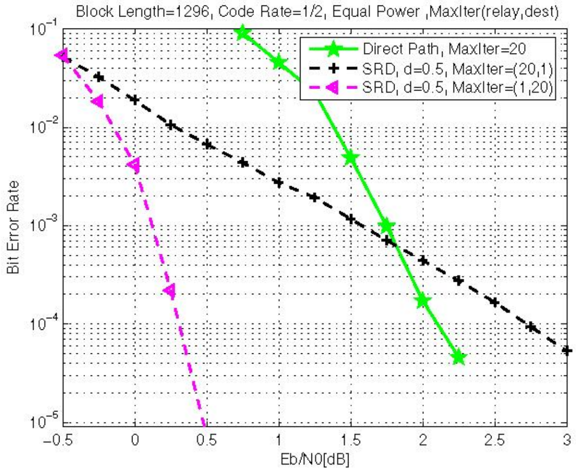


Fig. 5. Distributed and partial decoding, effect of number of iterations at the relay and the destination on the performance, (1296,1/2) LDPC code.

C. Distributed and Partial Decoding

In both DNEF and DEF schemes, it is assumed that the relay performs full decoding of the LDPC codeword until it converges to a correct codeword. Depending on the signal to noise ratio of the channel, this can take up to a hundred iterations, which results in long processing delays at the relay.

Figure 4 shows the percentage of bits that are in error after each decoding iteration for a $1296, 1/2$ LDPC code in a two node system (direct link). It can be seen that if the SNR is low, and the number of bits in error are high, more iterations need to be performed to correct the error. By increasing the SNR, less iterations are required to reach the correct codeword.

Since the relay is relatively close to the source, the signal is received with higher reliability at the relay. Therefore, the first few iterations correct most of the errors in the codeword.

Considering the DNEF scheme, since the LDPC code used at the SR and RD channels are equal, there is no need to wait for the decoding to converge to a correct codeword. This means that we can partially decode the codeword at the relay and retransmit it to the destination for full decoding. By 'Partially decoding', we mean decoding for a few iterations which corrects the errors in both information bits and the parity bits, eliminating the final step of the decoding that verifies the convergence to the correct codeword. Also instead of removing the parity bits and re-encoding the signal, we keep the corrected parity bits in the codeword, modulate the signal and re-transmit it to the destination. The destination receives two copies from the source and the relay and jointly decodes them. Our simulation results show that there is a major advantage in distributing the decoding process between the relay and the destination.

VI. Simulation Results

A. Performance of Distributed and Partial Decoding

We have analyzed the effect of distributed and partial decoding on cooperative communications. Assuming that we have limited processing time, limited number of iterations can be performed at the relay and the destination nodes.

Figures 5 and 6 show the effect of partial decoding with different number of iterations at the relay and at the destination (SRD), on the decoding performance. Compared with the basic single direct path, these figures show that if we have limited processing time/power, it is better to distribute the processing load between the relay and the destination. In the figures different bit error curves are shown for the DNEF strategy with partial decoding at the relay. The number of decoding iterations at the relay and at the destination is shown in the parenthesis in the figure legends.

In figure 5, comparing the simulation results of the direct path with 20 iterations at the destination with the cooperative communication link with one iteration at the relay and 20 iterations at the destination $(1, 20)$, shows that partially decoding with performing just one decoding iteration at the relay results in a major improvement in the decoding performance ($1.8dB$ gain in BER of 10^{-4}). Although the codeword at the relay still has many errors after one iteration.

Comparing the cases of (1,20) and (20,1) (Figure 5) and also (10,5),(5,10) (Figure 6) show that if we have limited number of iterations or equivalently limited latency to tolerate in the system, it is better to perform a few iterations at the relay and more iterations at the destination. This result is very appealing in practice since it translates into smaller processing delay and consumption of less power at the relay. This is especially appealing when the relays are battery operated devices that agree to relay the information of other users.

If we keep the number of iterations at the destination at a fixed number and increase the number of iterations at the

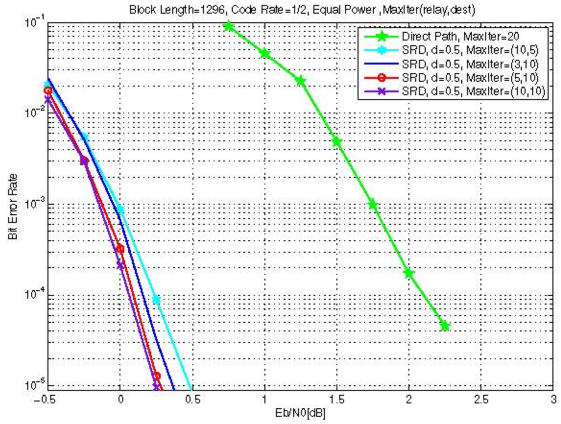


Fig. 6. Distributed and partial decoding, Effect of number of iterations at the relay and the destination on the performance, (1296,1/2) LDPC code.

relay (comparing the cases for (3,10), (5,10), (10,10)), we see that the first few iterations at the relay are very important, but the effect of the relay saturates as the number of iterations grows. This means that performing partial decoding with $2 \sim 3$ iterations at the relay and more iterations at the destination, gives us most of the benefits of the cooperation with much less decoding latency and processing power.

VII. Architectural Considerations

In [13], we presented a flexible decoder architecture for LDPC codes. This decoder supports a family of LDPC codes with block lengths of 648 to 1944 and code rates of $1/2$ to $5/6$. One of the advantages of partial decoding is that we can use the same decoder architecture that was designed for two node communication, in user cooperation scenarios. The only difference is that the parity bits are not discarded after the decoding. In partial decoding, fewer number of decoding iterations are performed on the codeword at the relay. Assuming that full decoding requires at least 10 decoding iterations to converge to a correct codeword, and assuming that we perform just 3 iterations for partial decoding at the relay, there is a savings of more than 70% in the decoding process. This translates to 70% savings in decoding latency because there is a linear relationship between the decoding latency and the number of decoding iterations. Also, it leads to 70% reduction in processing power at the relay.

Since partial and distributed decoding is part of the DNEF strategy, there is no need for the encoding hardware at the relay which also leads to savings in terms of the power/ area and latency.

VIII. Conclusion

In this paper we proposed the distributed and partial decoding scheme and showed its advantages over the decode and forward strategy such as reducing the latency and processing power at the relay over 70%. Also, we showed that just a few decoding iterations at the relay will result in about 2dB performance gain in the system. This strategy is very useful in practice for applications in which the relay is battery operated and does not want to spend most of its power processing the information of other users.

Acknowledgment

This work was supported in part by Nokia Corporation and by NSF under grants CCF-0541363, CNS-0551692, CNS-0619767, and EIA-0321266.

References

[1] E. van der Meulen, "Three terminal communication channels," *Adv. Appl. Prob.*, vol. 3, pp. 120–154, 1971.

[2] T. Cover and A. Gamal, "Capacity theorems for the relay channel," in *IEEE Transactions on Information Theory*, vol. 25, 1979, pp. 572 – 584.

[3] J. Laneman, G. Wornell, and D. Tse, "An efficient protocol for realizing cooperative diversity in wireless networks," in *IEEE International Symposium on Information Theory*, June 2001, p. 294.

[4] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity. Part I. System description," *IEEE Transactions on Communications*, vol. 51, no. 11, pp. 1927 – 1938, 2003.

[5] T. Hunter and A. Nosratinia, "Diversity through coded cooperation," *IEEE Transactions on Wireless Communications*, vol. 5, pp. 283 – 289, 2006.

[6] T. Rappaport, *Wireless Communications: Principles and Practice*. Prentice-Hall, Inc., 1996.

[7] M. M. Mansour and N. R. Shanbhag, "High-throughput LDPC decoders," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, pp. 976–996, Dec. 2003.

[8] S. Y. Chung, T. Richardson, and R. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Transactions on Information Theory*, vol. 47, pp. 657–670, Feb. 2001.

[9] P. Radosavljevic, A. de Baynast, and J.R. Cavallaro, "Optimized Message Passing Schedules for LDPC Decoding," in *IEEE 39th Asilomar Conference on Signals, Systems and Computers*, Nov. 2005, pp. 591–595.

[10] J. Chen, A. Dholakai, E. Eleftheriou, M. Fossorier, and X. Hu, "Reduced-complexity decoding of LDPC codes," *IEEE Transactions on Communications*, vol. 53, pp. 1288 – 1299, Aug 2005.

[11] M. Karkooti and J.R. Cavallaro, "Cooperative Communications Using Scalable, Medium Block-length LDPC Codes," in *IEEE Wireless Communication and Networking Conference* , 2008, submitted.

[12] H. V. Khuong and H.-Y. Kong;, "Performance analysis of cooperative communications protocol using sum-product algorithm for wireless relay networks," in *The 8th International Conference onAdvanced Communication Technology, ICACT 2006.* , 2006, pp. 2168–73.

[13] M. Karkooti, P. Radosavljevic, and J.R. Cavallaro, "Configurable high throughput irregular LDPC decoder architecture: tradeoff analysis and implementation," *IEEE 17th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, pp. 360–367, Sept. 2006.