# Reliability Exchange Schemes for Iterative Packet Combining in Distributed Arrays

Arun Avudainayagam, John M. Shea, Tan F. Wong and Xin Li
Department of Electrical and Computer Engineering
University of Florida
Email: {arun,lixin}@dsp.ufl.edu, {jshea,twong}@ece.ufl.edu

***Abstract*— We investigate reliability exchange schemes that allow a group of radios to act as a distributed antenna array in which radio links are used to exchange information between the various radios. We consider a scenario in which multiple nodes receive independent copies of the same message. Each node independently decodes its message and then participates in a process of "smart" information exchange with the other nodes. Each node transmits its estimates of the *a posteriori* probabilities of some set of bits, and these estimates are used as *a priori* information by other receivers. The *a priori* information is used to perform maximum *a posteriori* decoding on the received sequence. This process of decoding and information exchange can be repeated several times. Simulation results show that the performance can be significantly improved by careful selection of the bits for which information is exchanged.**

## I. Introduction

The use of antenna arrays is a common technique to achieve spatial diversity [1]. To achieve a significant gain in performance, the size of the antenna array must be several times the wavelength of the RF carrier. This constraint on the physical size of the antenna makes spatial processing using antenna arrays an unattractive choice in several scenarios. This is one reason why antenna arrays are mainly suggested for use in the uplink of a cellular system rather than the downlink. The small size of modern cellular phones cannot support the size of the antenna array required to achieve a significant improvement in performance.

In [2], Wong et. al propose a network-based approach to achieve spatial diversity without the use of physical antenna arrays for the simple case of a two node network. In this approach the different nodes in a network are considered as the elements of a large antenna array. Since the elements are not physically connected, this is referred to as a distributed array. All the nodes receive independent copies of a message. The nodes then make use of these independent copies in a smart manner to achieve diversity or antenna gain. Code combining [3] and iterative packet combining [4] have been studied for retransmission (ARQ) schemes. But these schemes rely on the presence of a feedback channel. In systems without feedback channels, a network-based approach as suggested in [2] is a practical alternative to achieve packet combining gains.

Since the arrays are not physically connected, performing maximal-ratio combining (MRC) [5] would be expensive in terms of communication overhead involved in disseminating information to all the nodes. For example, for an additive white Gaussian noise (AWGN) channel, using MRC would require sending the soft-demodulator outputs for each of the received symbols to all of the other nodes. This can drastically reduce the throughput of the system.

A suboptimal scheme that requires the exchange of significantly less information while still offering good performance was proposed for use with hybrid ARQ [6] and collaborative decoding [2]. In this scheme, each receiver uses a soft-input soft-output (SISO) maximum *a posteriori* (MAP) decoder [7]. Such a decoder typically takes as input an *a priori* probability for each information bit in addition to the received symbols and produces at the output *a posteriori* probabilities(APPs). In this paper, we consider log-MAP decoders for which the inputs and outputs are log-likelihood ratios(LLRs) of the specified probabilities. We refer to these input and output LLRs as *soft information*. The sign of the log-APP is the hard decision, and the magnitude of the log-APP is the reliability of that decision. Each node uses the bit reliabilities to determine which bits are unreliable (and thus need additional information from other nodes) or which bits are reliable (and thus should be shared with other nodes). The nodes then exchange soft information with other nodes in a process that we refer to as *reliability exchange*. The nodes use the received soft information as *a priori* information in a MAP decoder. This process can be repeated for a few iterations. The process of iterating between SISO decoding and reliability-exchange will be referred to as *collaborative decoding* since the nodes assist and work with each other to improve the performance. The information that passes between the different nodes will be referred to as the *overhead* in collaborative decoding.

This paper builds on [2] by investigating two different classes of reliability exchange schemes for multiple nodes (greater than two). The system topology that we consider is shown in Figure 1. A distant transmitter broadcasts a packet to a cluster of receiving nodes. A typical scenario would be a military application in which a battleship broadcasts a message to a platoon of soldiers on the mainland. In a commercial application a base station could communicate with a cluster of small mobile communicators. Because of the power limitation of the portable radios and the distance to the transmitter, ARQ schemes are not a feasible solution to combat decoding errors. The proposed schemes provide antenna gain while requiring a significantly lower overhead than maximal-ratio

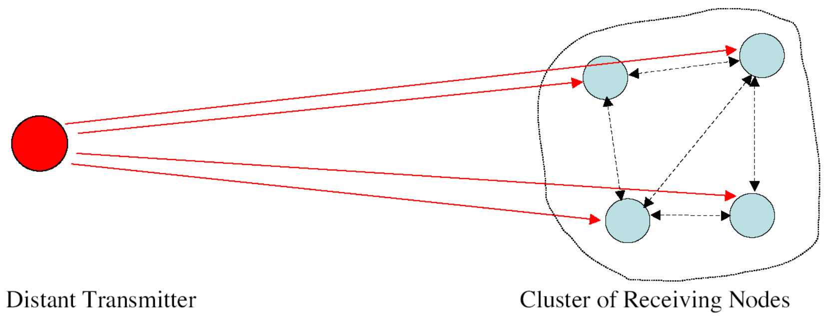


Fig. 1. System topology for collaborative decoding.

combining. We also introduce a suboptimal variant of this scheme that reduces the overhead drastically with very little loss in performance.

## II. Collaborative decoding through the reliability exchange of the least reliable bits

In this section, we provide results for an extension of the scheme proposed in [2]. For all the results in this paper, a rate 1/2 nonrecursive convolutional code with generator polynomials $1+D^2$ and $1+D+D^2$ is used to encode the information sequence. For convenience, we refer to this code as the $(5,7)$ code, where $5$ and $7$ are the octal representations of the generator polynomials. The encoded messages are transmitted over AWGN channels using binary antipodal signaling and are coherently demodulated. Each receiver decodes the received message using the BCJR [7] algorithm. Each node then requests reliability information for a certain percentage of the least reliable information bits by broadcasting the bit indices of those bits. Each node that receives the bit indices replies with its estimate of the soft information for those bits. The node that requested the information then uses these reliabilities as *a priori* information and runs the BCJR algorithm again. In [2] it is shown that for a packet size of approximately 1000 bits encoded with a $(5,7)$ convolutional code, collaborative decoding with two receivers provides performance very close to MRC at values of $E_b/N_0$ greater than 5 dB. Three iterations of collaborative decoding was performed by requesting soft information for $7.5\%$ of the least reliable information bits in each iteration. The reason for requesting the least reliable bits (LRBs) is that most of the bits that decode incorrectly have low reliability values. This fact is substantiated in Section III.

Using MRC would require exchanging all of the received coded symbols. The overhead in bits, denoted by $\Theta_{MRC}$ can be calculated as

$$\Theta_{MRC} = \frac{N}{R_c} \times q, \tag{1}$$

where $N$ is the size of the information message in bits, $R_c$ is the code rate and $q$ is the number of bits required to represent a (floating point) channel symbol. Note that (1) represents the overhead contribution of a single node. Using the collaborative decoding scheme mentioned above, the overhead contribution of a single receiver can be split into two parts. The first part consists of the bit indices that a receiver broadcasts to request the soft information of the LRBs, and the second part consists of the soft information that a node transmits each time it receives an LRB request from another node. Thus, the overhead for this scheme can be expressed as

$$\Theta_{LRB} = N_I \times a \times N \times (\lceil \log_2 N \rceil + (N_R - 1) \times q), \tag{2}$$

where $N_I$ is the number of iterations of collaborative decoding, $a$ is the fraction of information about which reliability information is requested, $N_R$ is the number of receivers involved in collaborative decoding and $\lceil x \rceil$ is the smallest integer greater than or equal $x$. The first term in the summation on the R.H.S of (2) accounts for the bit indices that need to be transmitted to request soft information, and the second term in the summation accounts for the bits required to send out the soft information (each node receives $N_R - 1$ LRB requests). Note that the size of the requests can be further reduced through source coding or by exploiting the time-correlation between the reliabilities of the bits in error [8]. Both (1) and (2) refer to the overhead per receiver. All the schemes in this paper will be compared using the overhead contribution per receiver.

Generally five bits are enough to represent a (floating point) channel symbol accurately [9], [10]. For a packet size of 900 bits, the overhead for MRC can be calculated using (1) as 9000 bits. Ten bits are required to represent each bit index in packet of $900$ bits, and if we perform three iterations of collaborative decoding with soft information of $5$ % of the LRBs being requested, the overhead is $2025$ bits for two nodes (using (2)). Thus we see that performing collaborative decoding reduces the overhead by $77.5$ % when compared to the MRC overhead.

Performing collaborative decoding with three iterations of $5$ % LRB exchange will be referred to as scheme LRB-1 for the rest of the paper. LRB-1 has collaborative decoding overhead of $22.5$ % of MRC overhead for a packet size of $900$ bits and a cluster size of two nodes. For the rest of the paper the overhead for collaborative decoding will be reported as a percentage with reference to the MRC decoding overhead. Note that the overhead per receiver for LRB-1 increases with the number of receivers. The overhead for LRB-1 for different number of nodes is shown in Table I. The reduction in overhead relative to MRC decreases with an increase in the

| Number of nodes | Overhead (bits) | % reduction (relative to MRC) |
|---|---|---|
| 2 | 2025 | 77.5 % |
| 5 | 4050 | 55.0 % |
| 10 | 7425 | 17.5 % |

TABLE I

OVERHEAD OF LRB-1 FOR DIFFERENT NUMBER OF NODES.

number of nodes.

The results in Figure 2 show the bit error probabilities for various LRB exchange scenarios versus $E_b/N_0$, the bit energy-to-noise density ratio. We note that the performance saturates for more than four receivers. This indicates that biasing the least reliable bits with a lot of *a priori* information from too many receivers will not improve the performance significantly. This is because there are some incorrectly decoded bits that may have relatively high reliabilities. This is again substantiated in the next section. When least-reliable bits are exchanged, the incorrect bits with high reliabilities may never be corrected regardless of how much information is provided for the LRBs.

An obvious method to improve the performance of LRB-1 is to increase the percentage of LRBs requested. From Figure 2, we see that requesting 10% of LRB reliabilities instead of 5% gives an improvement in performance of approximately 1 dB for a cluster of six collaborating nodes. However, this increases the collaborative decoding overhead, and our simulations show that the performance saturates for more than four receivers even in this case. Another disadvantage of requesting more information is that as the number of receivers increases, the time for information exchange also increases. Each receiver has to send out a set of bit indices requesting reliability information, and then all the other receivers have to respond. To coordinate this information exchange, a good MAC protocol will have to be designed. This latency would not be acceptable in certain applications.

A simple extension to LRB-1 is to transmit all the soft information via a broadcast channel and to have each node use *all* the received soft information, even if that node was not the original requester. Since the nodes other than the one that requested information also receive the soft information, they can make use of it as *a priori* information in their next round of SISO decoding. Thus, for the nodes that did not request the information, reliability information for a set of bits with random reliabilities is obtained. This scheme will be referred to as LRB-2. LRB-2 has the same overhead as LRB-1. The results in Figure 2 show that LRB-2 even outperforms LRB-1 with 10% LRB exchange. Further, LRB-2 does not suffer from the saturation problem like LRB-1. Hence, LRB-2 would be a better choice if exchanging LRBs was the scheme chosen to perform collaborative decoding.

The biggest disadvantage of this scheme is that the per-receiver overhead grows linearly with the number of receivers (see equation (2)). Thus, if the number of nodes is large, even requesting a very small percentage of LRB soft information

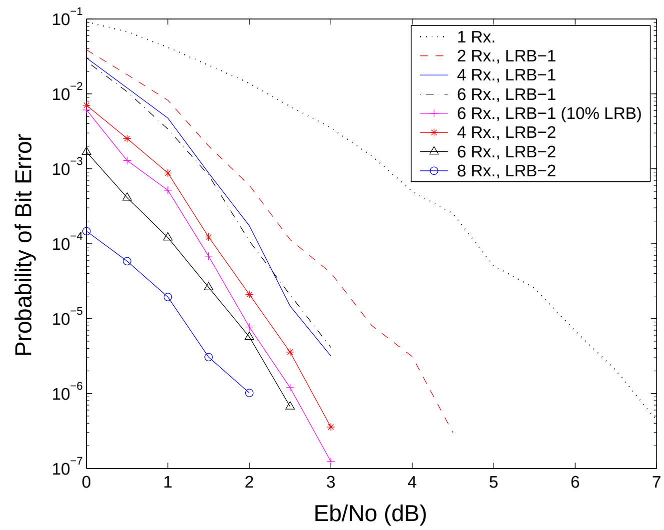


Fig. 2. Performance of two collaborative decoding schemes in which receivers request information for a set of least-reliable bits.

might cause the overhead to become larger than the MRC overhead. In the next section, an exchange scheme is presented that has an overhead that is independent of the number of receivers.

## III. COLLABORATIVE DECODING THROUGH THE RELIABILITY EXCHANGE OF THE MOST RELIABLE BITS

One way to significantly reduce the overhead is to prevent a node from transmitting soft information more than once. From (2), we see that for block sizes of approximately 1000 and more than ten receivers, multiple transmissions of soft information contributes towards more than 82% of the overhead per receiver. Suppose that, after SISO decoding, each receiver selects a certain set of bits and broadcasts the reliabilities of these bits to the other nodes. It is important to ensure that the nodes broadcast "good" reliability information, i.e., reliability information about bits that are decoded correctly. The critical step in this scheme is to determine the set of bits for which a node will broadcast the soft information. Since each node only sends out soft-information only once, the collaborative decoding overhead per receiver is given by

$$\Theta_{MRB} = N_I \times a \times N \times (\lceil \log_2 N \rceil + q). \qquad (3)$$

Note that for this scheme, unlike the LRB-based schemes, the overhead per receiver is independent of the number of receivers.

If a node broadcasts the soft information for a bit that was decoded incorrectly, using this value as *a priori* information would degrade the performance of the other nodes. The motivation behind our approach for selecting bits comes from observing the density functions of the reliabilities associated with correctly and erroneously decoded bits. Figure 3 shows the density function of the reliabilities for a $(5, 7)$ convolutional code with a block size of 900 bits.

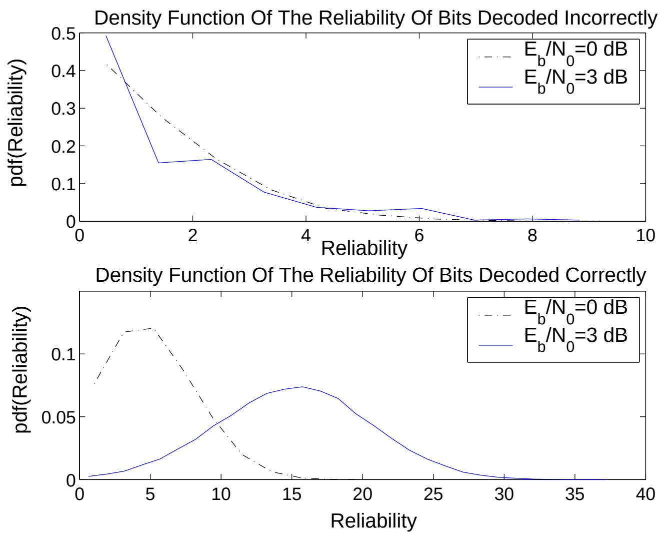


Fig. 3. Reliability density functions associated with correctly and incorrectly decoded bits.

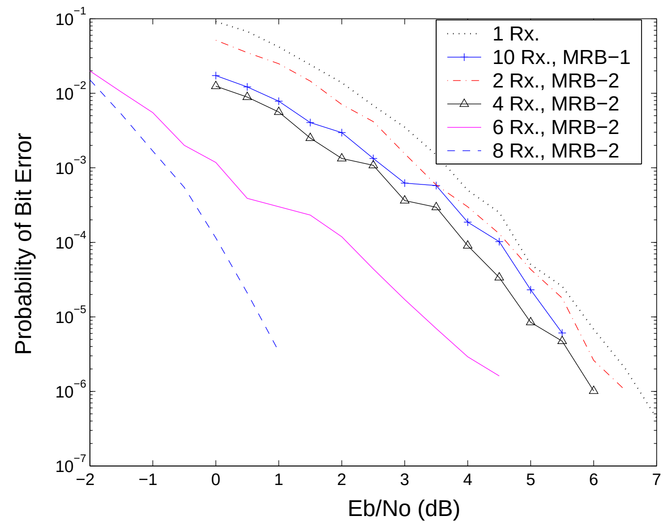


Fig. 4. Performance of two collaborative decoding schemes in which receivers broadcast information about a set of most-reliable bits.

We note that the trend followed by the density functions is in agreement with the theoretical expressions given in [11] and [12]. By observing the conditional density function of the reliabilities of the bits given that they were decoded incorrectly, we note that there are very few bits that have very high reliabilities and yet are decoded incorrectly. We observe that at an $E_b/N_0$ of 0 dB, the maximum value of the reliability of a bit that decodes incorrectly is about half of the maximum value of the reliability of a bit that decodes correctly. For values of $E_b/N_0$ greater than 3 dB, more than 50% of the bits that decode correctly have reliabilities greater than the maximum reliability of the incorrectly decoded bits. Hence, if a node broadcasts a small percentage of its most reliable bits (MRBs), it is very likely to send out "good" soft information. These bits will correspond to a set of bits with random reliabilities at the other nodes. Performing three iterations of 10% MRB reliability exchange will be referred to as scheme MRB-1. The collaborative decoding overhead (per receiver) of MRB-1 is calculated, using (3), as 45% that of MRC. Though reliability information is exchanged for more bits than in LRB-1 and LRB-2, the overhead is still smaller than in LRB-1 and LRB-2 for more than five nodes.

Our simulations showed that the performance improvement is less than 2 dB even with ten nodes when compared to the performance of a single receiver. This is shown in Figure 4. The smaller performance improvement can be attributed to the set of bits that are broadcast in each iteration. At the end of the first SISO decoding, reliabilities of 10% of the most reliable bits are broadcast. Since we are biasing certain bits with "good" *a priori* information, at the end of the next SISO decoding, the reliabilities for these bits will become large, and it is very likely that these bits will lie in the 10% MRB set. So reliabilities of these bits will be broadcast in the next iteration. But since the other nodes have already received the information about these bits, biasing them with more *a priori* information will not improve the performance significantly. This can be observed in Figure 5 in which we show the bit indices broadcast in each iteration for one packet of 900 bits. An asterisk on a bit position implies that reliability information about that bit was either requested (LRB-1) or transmitted (MRB schemes). We see that for MRB-1, a very large portion of bits are broadcast again in every iteration. In three iterations, the reliabilities of 102 bits are broadcast again among the total of 270 bits transmitted. This constitutes around 37% of the total bits sent. As the value of $E_b/N_0$ increases, there are fewer bits in error, and in order to improve the performance, these erroneously decoded bits need to be biased with reliable *a priori* information. If a good percentage of the bits are repeated, there will be a low probability that *a priori* information will be received for all of the bits that are in error.

A simple method to eliminate this problem is to give the nodes memory to remember the set of bits for which soft information is transmitted or received. This ensures that bits that are already likely to have good reliabilities after one iteration do not get biased with more *a priori* information in the next iteration. Other bits are now given an opportunity to receive reliability information. This scheme, which is just MRB-1 with memory, will be referred to as MRB-2. In MRB-2, each node sorts its bits in ascending order of reliability after the first SISO decoding. Then each receiver broadcasts 10% of the MRBs for which soft information was not transmitted by any node in the previous iterations. Thus, in each iteration, reliability information is received for a new set of bits. This is illustrated in Figure 5. In MRB-2, there are no bits for which soft information is transmitted in more than one iteration. The performance of MRB-2 is shown in Figure 4. If in any of the iterations, a node is not able to find a bit about which *a priori* information has not been transmitted earlier, it does not send

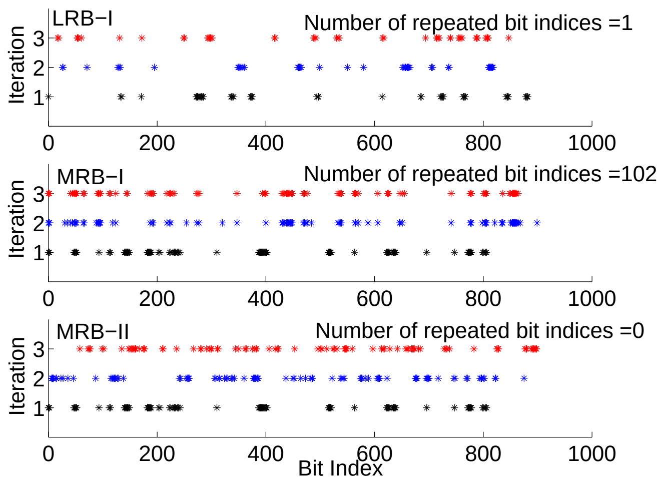


Fig. 5. Bit Indices of reliabilities exchanged as a function of iteration.

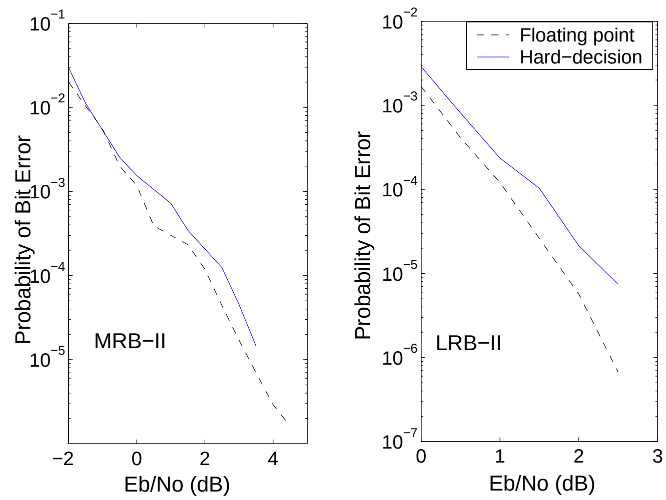


Fig. 6. Comparison between exchange of hard decisions and floating-point values for collaborative decoding with six receivers.

out any reliabilities. Thus, the overhead in MRB-2 is less than or equal to the overhead in MRB-1, but the performance of MRB-2 is much better than that of MRB-1.

Note that adding memory to LRB-1 will not improve the performance significantly. This is because in each iteration, *a priori* information biases the least reliable bits and their reliability increases after SISO decoding. Thus, in the next iteration a new set of bits will constitute the set of LRBs. Hence, there is only a negligible overlap in the set of LRBs in each iteration. This can be observed in Figure 5, in which LRB-1 has just one bit that is repeated in three iterations.

A good suboptimal variant of MRB-2 sends hard-decisions of the MRBs instead of the soft decisions. This reduces the overhead for transmitting soft information from $q$ bits (see equation (3)) per symbol to one bit per symbol. Thus, for MRB-2 with three iterations of 10% reliability exchange, the collaborative decoding overhead is only 33% for a packet of 900 bits. For a reasonably large number of receivers, the hard decisions from different receivers form *a priori* information that is sufficient to bias the information bits to produce correct decisions at the output of the SISO decoder. The performance of this scheme for six receivers is illustrated in Figure 6.

Note that this technique can be extended to any of the schemes discussed earlier. The performance of LRB-2 with hard-decisions is also shown in Figure 6. We see that a loss of approximately $0.5$ dB can be expected for the suboptimal scheme when compared to the original scheme.

The schemes that work with the MRBs also require less-complex channel access techniques. If the number of nodes are fixed, a simple round robin of all the nodes can be used to allow them to broadcast reliabilities of a certain percentage of their MRBs. For dynamically formed *ad hoc* networks, a cluster head could be chosen that assigns the order in which the nodes broadcast the reliabilities.

A performance comparison of the MRB scheme and the LRB scheme discussed in the previous section is given in Figure 7 for a cluster of six nodes. It is seen MRB-2 outperforms LRB-1 for low values of $E_b/N_0$ and LRB-1 performs better than MRB-2 for high values of $E_b/N_0$. An improvement in performance is obtained when bits that are decoded incorrectly get good *a priori* information. At low values of $E_b/N_0$, it is highly probable that some of the bits that have very low reliabilities are bits that have been decoded correctly. This can be observed in Figure 3, in which at 0 dB there is a good percentage of correctly decoded bits with soft output values less than the maximum soft output value of an incorrectly decoded bit. Thus, requesting information about a certain percentage of the LRBs will include bits that already decode correctly but that have low soft-output values. Biasing these correct bits with *a priori* information will not provide significant gains. Our simulations show that the majority of the errors that are not corrected by this scheme are bits about which reliability information was not requested. But at higher values of $E_b/N_0$, there are fewer bits that decode correctly and have low reliabilities (see equation figure 3), and so the least reliable bits are almost always in error. Hence for large $E_b/N_0$, all the bits biased with *a priori* information are bits that were originally in error at the output of the decoder. This is the reason why LRB-1 performs better than MRB-2 for six receivers at high values of $E_b/N_0$. In MRB-2, soft outputs of a certain percentage of MRBs are broadcast and with high probability these bits are the correctly decoded bits as explained in Section III. Thus, at one of the other nodes, good reliabilities are received for a set of bits with random reliabilities. There may or may not be an incorrectly decoded bit in each of these bit positions. However, for a large number of receivers, it is likely that many of the bits in error are covered. With 10% of MRB exchange and more than eight receivers, it is likely that information will be exchanged for almost all the bits in a block of 900 bits, and hence the performance of MRB-2 is better than LRB-2 for more than eight receivers.

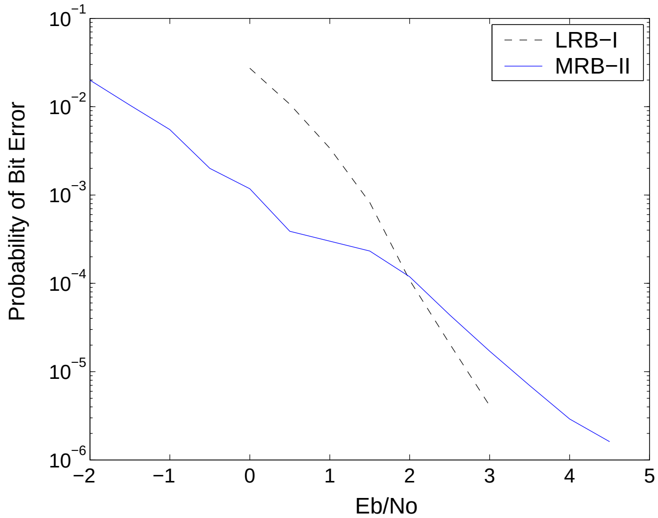


Fig. 7. Performance of LRB-1 and MRB-2 schemes with 6 nodes.

## IV. CONCLUDING REMARKS

In this paper, collaborative decoding schemes were investigated for clusters of two or more nodes. Two different classes of reliability exchange schemes are proposed. In one class of schemes, the nodes request soft information for a certain percentage of their least reliable bits (LRBs). The disadvantage of the LRB-based schemes is that the overhead per receiver grows linearly with the number of receivers. In the other class of reliability exchange schemes, nodes send out soft information about a small set of their most reliable bits(MRBs). These schemes have a fixed per-receiver overhead that is independent of the number of nodes. In clusters with eight or more nodes, antenna gain of more than 5 dB can be achieved using the LRB-based schemes and a gain of more than 7 dB can be achieved using the MRB-based schemes. We have also shown that suboptimal schemes can be designed that have extremely low overheads and a tolerable loss in performance. Thus, collaborative decoding provides an efficient way of improving performance in systems without a feedback channel.

## V. ACKNOWLEDGMENTS

This work was supported by the DoD Multidisciplinary University Research Initiative (MURI) program administered by the Office of Naval Research under Grant N00014-00-1-0565 and by the Office of Naval Research under Grant N00014-02-1-0554.

## REFERENCES

[1] R. T. Compton, *Adaptive Antennas: Concepts and Performance*. 1st. ed., Prentice Hall, 1988.
[2] T. F. Wong, X. Li, and J. M. Shea, “Iterative decoding in a two-node distributed array,” in *Proc. 2002 IEEE Military Communications Conference (MILCOM)*, vol. 2, (Anahiem, CA), pp. 704.2.1–5, Oct. 2002.
[3] D. Chase, “Code combining-A maximum-likelihood approach for combining an arbitrary number of noisy packets,” *IEEE Trans. Commun.*, vol. COM-33, pp. 385–393, May 1985.
[4] D. N. Doan and K. R. Narayanan, “Iterative packet combining schemes for intersymbol interference channels,” *IEEE Trans. Inform. Theory*, vol. 50, pp. 560–570, Apr. 2002.
[5] J. G. Proakis, *Digital Communications*. 4th.ed., McGraw-Hill, 2000.
[6] J. M. Shea, “Reliability-based hybrid arq,” *IEEE Electronics Letters*, vol. 38, pp. 644–645, June 2002.
[7] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rates,” *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.
[8] A.Roongta and J. M. Shea, “Reliability based hybrid ARQ using convolutional codes,” in *Int. Conf. Commun. (ICC)*, (Anchorage, AK), Submitted to, May 2003.
[9] B. Blanchard, “Quantization effects and implementation considerations for turbo decoders,” Master’s thesis, University of Florida, 2002.
[10] G. Montorsi and S.Benedetto, “Design of fixed-point iterative decoders for concatenated codes with interleavers,” *IEEE J.Select. Areas Commun.*, vol. 19, pp. 871–882, May 2001.
[11] D. Agrawal and A. Vardy, “Generalized minimum distance decoding in Euclidean space:Performance analysis,” *IEEE Trans. Commun.*, vol. 46, pp. 60–83, Jan. 2000.
[12] L. Reggiani and G. Tartara, “Probability density functions of soft information,” *IEEE Commun. Letters*, vol. 6, pp. 52–54, Feb. 2002.