Nonlinear Trellis Codes for Binary-Input Binary-Output Multiple-Access Channels With Single-User Decoding

Miguel Griot, Andres I. Vila Casado, Wen-Yen Weng, Herwin Chan, *Member, IEEE*, Jiadong Wang, *Student Member, IEEE*, and Richard D. Wesel, *Senior Member, IEEE*

***Abstract*— This paper presents a practical technique that uses Ping's interleave(r)-division multiple access and single-user decoding to provide uncoordinated access for a family of binary-input binary-output multiple-access channels (MACs) including the OR-MAC where users' binary transmissions are combined with the logical OR operation. Information theoretic calculations provide the achievable sum-rates and optimal ones densities for these MACs. Because the required ones densities are significantly less than 50%, new nonlinear trellis code analysis and design techniques are introduced to provide the needed codes. Union bound techniques that predict the performance of these codes are also presented. Simulation results and a working FPGA implementation verify the performance and feasibility of the proposed nonlinear codes and overall multiple access scheme.**

***Index Terms*— Multiuser channels, single-user-decoding, nonlinear trellis codes, binary asymmetric channel, Z channel, multiple access channel, binary multiplier channel.**

I. Introduction

THERE have been many approaches to providing multiple users access to the same channel. Access can be coordinated or uncoordinated. Coordinated multiple access schemes include time-division (TDMA), frequency-division (FDMA), code-division (CDMA), and rate-splitting [1]. A joint trellis-code design for all users has been proposed in [2], but this also requires coordination as a distinct channel code is assigned to each user.

When coordination is either not possible or not convenient, uncoordinated multiple access schemes such as Aloha, slotted Aloha, carrier-sense multiple access (CSMA), and CSMA with carrier detection (CSMA-CD) are employed. However, these schemes do not provide a clear QoS in terms of delay or delay jitter. One recent approach for uncoordinated multiple-access that does not introduce delay jitter is Interleave(r)-Division Multiple-Access (IDMA) [3], which uses interleaving to distinguish among signals from different users.

This work explores the applicability of the IDMA approach to a family of with Binary-Input, Binary-Output Multiple Access Channels (BIBO-MACs). In particular, we consider the OR Multiple Access Channel (OR-MAC), or its isomorphic channel, the Binary Multiplier Channel [4, Example 15.3.2], as a target application for IDMA. Completely uncoordinated transmissions using IDMA and simple decoding that treats all signals except the desired signal as noise can theoretically achieve about 70% of the sum capacity over the OR channel for any number of users. Thus by sacrificing some portion of the sum rate, the IDMA approach provides a significant reduction in complexity over coordinated transmission or joint decoding approaches, making it practically attractive. We also explore a more general subset of the BIBO-MAC, which we call the OR-with-Interference MAC (ORI-MAC).

For OR- and ORI-MACs, IDMA requires channel codes with low ones densities. This paper investigates Nonlinear Trellis Codes (NLTCs) [2] to provide the required ones densities and to permit low-complexity decoding. Additionally, NLTCs can support IDMA even for large numbers of users. Turbo solutions, which more closely approach the sum-capacity at the cost of more latency and complexity in the decoding, have also been explored in our related paper [5].

Section II reviews uncoordinated multiple access in the BIBO-MAC, and in particular the OR-MAC and ORI-MAC. Section III presents an NLTC design technique for this application. Section IV analyzes the performance of these codes for large numbers of users, presenting an analytical tool to choose the proper number of states for the trellis code. Section V introduces a transfer-function bound for NLTCs operating on the Z-Channel and a separate bound that applies to any binary asymmetric channel (BAC). Section VI presents performance results, and Section VII concludes the paper.

II. The BIBO-MAC Model

In the BIBO-MAC there are N users each transmitting a binary symbol (bit). For one transmission time, denote the

Paper approved by T. M. Duman, the Editor for Coding Theory and Applications of the IEEE Communications Society. Manuscript received March 15, 2007; revised October 23, 2007.

This work was presented in part at the 2006 IEEE International Symposium on Information Theory in Seattle, Washington and the 2006 Workshop on Information Theory and its Applications in San Diego, California. This work was supported by the Defense Advanced Research Project Agency under SPAWAR Systems Center San Diego Grant N66001-02-1-8938.

M. Griot is with Qualcomm Inc., San Diego, CA 92121 USA (e-mail: mgriot@qualcomm.com).

A. I. Vila Casado is with Mojix Inc., Los Angeles, CA 90025 USA (e-mail: andres@mojix.com).

W.-Y. Weng is with Ralink Technology, Taiwan (e-mail: wenyen.weng@ralinktech.com).

H. Chan is with the Pharmaco-Kinesis Corporation, Inglewood, CA 90304 USA (e-mail: herwin.chan@pharmaco-kinesis.com).

J. Wang and R. D. Wesel are with the Electrical Engineering Department, University of California, Los Angeles, CA 90095 USA (e-mail: wjd@ee.ucla.edu; wesel@ee.ucla.edu).

Digital Object Identifier 10.1109/TCOMM.2011.01.xxxxx

transmitted bits as $\{x_1, \cdots, x_N\}$ and the received binary symbol as y. Denote by ψ_m the conditional probability of receiving a 0 given that the sum of the transmitted bits is m:

$$\psi_m = P\Big(y=0\Big|\sum_{i=1}^{i=N} x_i = m\Big), \quad m \in \{0, \cdots, N\}. \tag{1}$$

The set of parameters $\{\psi_m : m = 0, ..., N\}$ depend on the particular BIBO-MAC.

A. Uncoordinated Access to the BIBO-MAC

This paper applies IDMA to the BIBO-MAC in a completely uncoordinated manner. Thus, time division is not allowed. For simplicity, we assume all the users are transmitting all the time at the same rate[1] and all the users transmit with the same ones density $p = P(x_i = 1)$, for all $i = 1, \cdots, N$. Under these assumptions, for a specified ones density p, the achievable sum of the rates of all the N users, which we will call the sum-rate $R_N(p)$, is upper bounded as follows:

$$\begin{aligned} R_N(p) \le & H\left(\sum_{m=0}^{N}\binom{N}{m}p^m(1-p)^{N-m}\psi_m\right) \\ & - \sum_{m=0}^{N}\binom{N}{m}p^m(1-p)^{N-m}H(\psi_m), \end{aligned} \tag{2}$$

where $H(\cdot)$ is the binary entropy function. To maximize the sum-rate, the optimal ones density is then:

$$p^{opt} = \text{argmax}_{p\in[0,1]}\{R_N(p)\}. \tag{3}$$

Depending on the particular values ψ_m, the sum-rate can take values from 0 (for the case $\psi_m = 1/2, \forall m$) to 1 (the upper bound since the output is binary). This approach may not be capacity achieving, since higher rates might be achieved with time division. However, we will investigate cases where time division is not necessary to achieve capacity or where the increase in rate possible with time division is small.

B. BIBO-MAC With Single-User Decoding

The BIBO-MAC sum-rate capacity may be achieved with joint decoding of all the transmitted sequences. However, joint decoding can be very complex, especially for a large number of users. In high-speed applications where joint decoding is unavailable for complexity reasons, Single-User Decoding (SUD) must be used. With SUD, each user treats all but the desired signal as noise, transforming the BIBO-MAC into a Binary Asymmetric Channel (BAC) for each user with the following cross-over probabilities:

$$\alpha = P(y{=}1|x{=}0) \tag{4}$$

$$= 1 - \sum_{m=0}^{N-1}\binom{N-1}{m}p^m(1-p)^{N-1-m}\psi_m, \tag{5}$$

$$\beta = P(y{=}0|x{=}1) \tag{6}$$

$$= \sum_{m=0}^{N-1}\binom{N-1}{m}p^m(1-p)^{N-1-m}\psi_{m+1}. \tag{7}$$

[1]In fact, the essential results hold for users with different rates and correspondingly different ones densities, but considering these cases in detail would unnecessarily complicate the exposition.

$R_N^{SUD}(p)$, the achievable sum-rate for N users with SUD and a fixed common ones density p behaves as follows:

$$\frac{R_N^{SUD}}{N} \le H\big((1{-}p)(1{-}\alpha){+}p\beta\big){-}(1{-}p)H(\alpha){-}pH(\beta)\,. \tag{8}$$

C. The OR-MAC

In the OR-MAC, if all users transmit a 0, then the channel output is a 0. However, if one or more users transmit a 1, then the channel output is a 1. The OR channel can be used as a simple communications model that describes a multiple-user local area network optical channel with non-coherent combining. For short distances, the effect of noise can be considered negligible. A 1 is transmitted as light and a 0 is transmitted as no light. If any user transmits light, light is received. Only when all users do not transmit light, a 0 is received. In this simple model it is assumed that there is no destructive interference between users.

The OR-MAC is a particular case of the BIBO-MAC channel presented above where:

$$\text{OR-MAC} : \psi_m = \begin{cases} 1 & \text{for} \quad m = 0, \\ 0 & \text{for} \quad m = 1, \cdots, N. \end{cases} \tag{9}$$

Applying (9) to (2) yields $R_N \le H\left((1-p)^N\right)$ so that the maximum possible sum rate of 1 is achieved with

$$\text{OR-MAC} : p^{opt}(N) = 1 - (1/2)^{1/N}. \tag{10}$$

Note that the achievable sum-rate is 1 regardless of the number of users N. This means that if joint decoding is employed, completely uncoordinated transmission on the OR-MAC is theoretically possible with the same efficiency as TDMA, for any number of users.

When single-user decoding is used over the OR-MAC, each user perceives a particular case of the BAC, commonly known as the Z-Channel, where:

$$\alpha = 1 - (1-p)^{N-1}, \quad \beta = 0. \tag{11}$$

The achievable sum-rate becomes:

$$R_N^{SUD}(p) = N \cdot \Big[H\big((1-p)^N\big) - (1-p)H(\alpha)\Big]. \tag{12}$$

An interesting property of the OR-MAC is that with single-user decoding its maximum theoretical sum-rate

$$R_N^{SUD} = \max_p R_N^{SUD}(p) \tag{13}$$

monotonically decreases only to $\ln 2 \simeq 0.6931$ as the number of users N increases. This is a relatively small loss in rate for the substantial reduction in complexity. Also, the optimal ones density is practically the same as the ones density for joint decoding. Namely:

$$p^{opt}(N) \to 1 - (1/2)^{1/N}, \text{ for SUD as } N \to \infty. \tag{14}$$

A sketch of the proof of the asymptotic sum rate and the asymptotically optimal ones density for SUD on the OR-MAC is as follows: First, we prove that R_N^{SUD} is monotonically decreasing. Let $p(N{+}1)$ be the optimal ones density for $N{+}1$ equal-rate users under single-user decoding. Compute the possibly suboptimal ones density $\tilde{p}(N)$ for N equal-rate users as the one that satisfies $(1-\tilde{p}(N))^N = (1-p(N+1))^{N+1}$.

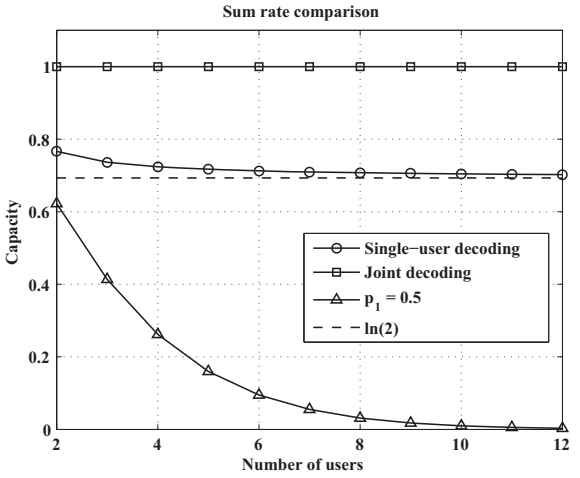


Fig. 1. Maximum OR-MAC sum-rates (in bits) as a function of the number of users for single-user decoding and joint decoding. Also shown is the sum rate for single-user decoding with a ones density of 0.5.

Using Eq. 12, note that $R_N^{SUD}\big(\tilde{p}(N)\big) > R_{(N+1)}^{SUD}\big(p(N+1)\big)$. Thus the sum rate is monotonically decreasing because a possibly sub-optimal ones density for N users leads to a higher symmetric sum rate than the optimal ones density for $N+1$ users.

To prove that the limit is $\ln 2$, prove that for a fixed number of users N, $R_N^{SUD}(p)$ is quasi-concave as a function of p, and since $R_N^{SUD}(0) = R_N^{SUD}(1) = 0$, there exists one (and only one) local maximum which is the global maximum. Then prove that the conjectured $p^{opt}(N)$ of Eq. (14) tends to the local maximum when $N \to \infty$. Finally, using the right-hand expression of Eq. (14) in Eq. (12), and letting $N \to \infty$ reveals that the limit is $\ln 2$.

Fig. 1 shows the maximum theoretical sum-rate using both joint decoding (which is always 1) and single-user decoding (which decreases to $\ln 2$). It also shows the maximum theoretical sum-rate of single-user decoding using a ones density of $p = 0.5$, which rapidly decreases to zero as the number of users increases. The poor performance of the $p = 0.5$ ones density demonstrates that codes with low ones densities are required for this application.

D. The OR-With-Interference MAC

We now extend the OR-MAC to handle the possibility of destructive interference, as might occur in optical multiple access. As with the OR channel, when all users transmit a 0, a 0 is received. Also, when only one user transmits a 1, a 1 is received. However, when more than one user transmits a 1, there is a certain probability (associated with a destructive interference event) that a 0 is received. This probability is always less than 1/2 and decreases as the number of ones transmitted increases. We call this the OR-with-Interference MAC (ORI-MAC).

Following the notation introduced in Section II-A, the ORI-MAC can be expressed as a general BIBO-MAC with the following constraints:

$$\text{ORI-MAC}: \begin{cases} \psi_0 = 1, \psi_1 = 0, \\ 1/2 \geq \psi_m \geq \psi_{m'}, \quad \forall m' \geq m \geq 2 \end{cases} \tag{15}$$

As a specific example of an ORI-MAC consider the Coherent Interference MAC (CI-MAC), for which:

$$\text{CI-MAC}: \begin{cases} \psi_0 = 1, \psi_1 = 0, \\ \psi_m = P\Big(\Big|\sum_{i=1}^{m} e^{j\theta_i}\Big|^2 < \sigma\Big), \quad \forall m \geq 2, \end{cases} \tag{16}$$

where $\theta_i \sim U[0, 2\pi)$ are random variables with uniform distribution, and σ is a threshold which will be considered 1/2 in this work.

The maximum achievable sum rate of the ORI-MAC, with coordinated time division, is 1. To see this, note that if the outputs of all but one user are set to 0, then the maximum achievable rate for that user is 1. Hence, with coordinated time division, any combination of rates with sum-rate equal to 1 can be achieved.

For the ORI-MAC without time division, it can be proven that the sum-rate capacity, for both joint and single-user decoding, is lower bounded by a strictly positive number regardless of the number users, as stated in the following theorems.

Theorem 1: Using a ones density of the form

$$p(N) = 1 - \delta^{1/N}, \tag{17}$$

the achievable sum rate on the ORI-MAC without time division is lower bounded by:

$$R_N \geq \max_{\delta \in [1/2,1]} \Big\{ H\Big(\delta + \psi_2 f(\delta)\Big) - f(\delta) H(\psi_2) \Big\}, \tag{18}$$

for any number of users N, where $f(\delta) = 1 - \delta + \delta \ln \delta$.

Theorem 2: Using a ones density of the form shown in (17), the achievable sum rate on the ORI-MAC without time division and with single-user decoding is lower bounded by:

$$R_N^{SUD} \geq \max_{\delta \in [1/2,1)} \Bigg\{ \ln \delta \Bigg[\Big(H\left(\psi_2(1-\delta)\right) - H\left(g(\delta)\right) \Big) + \Big(\psi_2(1-\delta) - g(\delta)\Big) \log_2 \left(\frac{g(\delta)}{1 - g(\delta)} \right) \Bigg] \Bigg\} \tag{19}$$

for any number of users N, where $g(\delta) = \delta + \psi_2 f(\delta)$.

The proofs are sketched as follows: First prove that the values of the sum rates $R_N(p)$ in (2) and $R_N^{SUD}(p)$ in (8) with $p(N)$ as in (17) are decreasing with N for fixed ψ_m's. Given a certain value of ψ_2, consider the worst case scenario $\psi_m = \psi_2$, $\forall m \geq 2$. Then use (17) in (2), and let $N \to \infty$ to prove Theorem 1. Use (17) in (5,7) and (8), and let $N \to \infty$ to prove Theorem 2.

For example, part of the argument for Theorem 1 is to evaluate the following expression from (2):

$$H\left(\sum_{m=0}^{N} \binom{N}{m} p^m (1-p)^{N-m} \psi_m \right) \tag{20}$$

$$= H\left(\delta + \psi_2 \left[1 - \delta - \delta N \left(\delta^{-1/N} - 1 \right) \right] \right) \tag{21}$$

$$\to H\left(\delta + \psi_2 \left[1 - \delta + \delta \ln \delta \right] \right) \tag{22}$$

where

$$\lim_{N \to \infty} N \left(\delta^{-1/N} - 1 \right) = -\ln \delta \,. \tag{23}$$

Fig. 2 shows achievable sum-rates (without time division) for the ORI-MAC for $\psi_m = \psi_2$ for all $m \geq 2$ (the worst possible channel given ψ_2) as a function of ψ_2 for joint

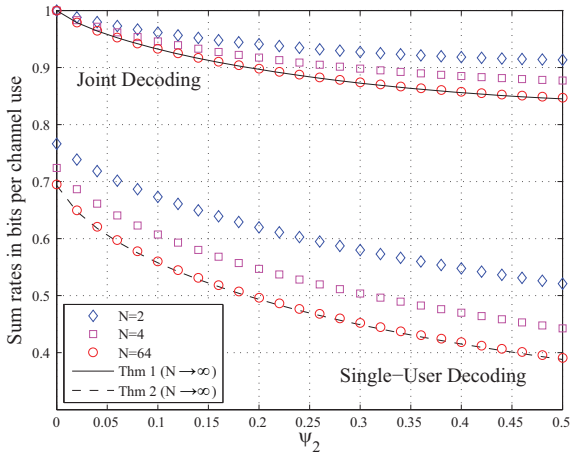


Fig. 2. Achievable sum-rates without time division for the ORI-MAC for $\psi_m = \psi_2$ for all $m \geq 2$ as a function of ψ_2 for joint decoding and for single-user decoding. Curves are shown for $N = 2,\ 4,$ and 64 users as well as for the limiting case of $N \to \infty$ (according to Thms. 1 and 2).

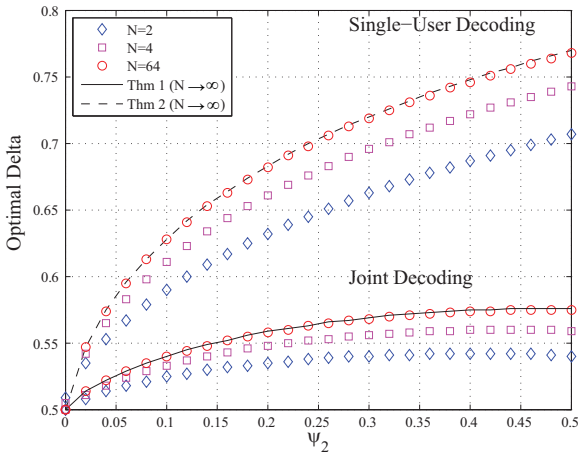


Fig. 3. Optimal δs for the ORI-MAC for $\psi_m = \psi_2$ for all $m \geq 2$ as a function of ψ_2 for joint decoding and for single-user decoding. Curves are shown for $N = 2,\ 4,$ and 64 users as well as for the limiting case of $N \to \infty$ (according to Thms. 1 and 2).

decoding using (2) and for single-user decoding using (8). Curves are shown for $N = 2,\ 4,$ and 64 users as well as for the limiting case of $N \to \infty$ (according to Thms. 1 and 2).

The asymptotic lower bounds of Thms. 1 and 2 are tight for 64 or more users. Note that as ψ_2 goes to 0, and hence the ORI-MAC tends to the OR-MAC, the sum-rate asymptotic lower bounds tend to OR-MAC achievable sum-rates, both for joint decoding and single-user decoding.

Fig. 3 shows the optimal δs for the ORI-MAC without time division for $\psi_m = \psi_2$ for all $m \geq 2$ as a function of ψ_2 for joint decoding and for single-user decoding. Curves are shown for $N = 2,\ 4,$ and 64 users as well as for the limiting case of $N \to \infty$ (according to Thms. 1 and 2). Note that as ψ_2 goes to 0, δ tends to $1/2$ regardless of the number of users for both joint decoding and single-user decoding, resulting in the optimal δ for the OR-MAC. Also note that the δ that maximizes the asymptotic lower bound is relatively close to the optimal delta for a finite N both for joint decoding and for single-user decoding.

III. NLTC With Controlled Ones Density

With IDMA, every user has the same channel code, but each user's encoder output bits are permuted using a randomly drawn interleaver, unique with extremely high probability. The receiver is assumed to know the interleaver of the desired user. With IDMA in the OR-MAC, a receiver sees the desired signal corrupted by a memoryless Z-channel. We performed simulations comparing an NLTCM code under two channels: 1) a 6-user OR-MAC (or ORI-MAC) channel using IDMA and 2) the equivalent Z-channel (or BAC) that the receiver would see if the errors were not generated by other users' codewords but by random errors. The performance was the same. Thus, in the context of IDMA, the remaining challenge is the design of good codes with the desired ones densities for the binary asymmetric channel and for the Z-Channel.

Papers appearing since the 1950's have addressed the problem of designing codes with $p = 0.5$ for the binary asymmetric channel, and in particular the Z-Channel. See [6] for a unified discussion of such codes and [7] for recent advances in this field.

Two recent papers, [8] and [9], addressed the problem of designing codes with a nonuniform ones density in the output by using sparse LDPC codes over large finite fields, *i.e.* using symbols from $GF(q)$ in the parity-check matrix. However, this solution requires a much more complex decoder than binary LDPC codes. Moreover, in the application considered in this work, the required low ones densities would lead to very large values of q. Also, the possible values of q limit the granularity of the output distribution.

In this section we present a design technique for trellis codes with an arbitrary ones density for the binary asymmetric channel and for the Z-Channel specifically. We use a conventional shift register for a rate-$1/n_0$ feed-forward encoder in order to determine the state transitions of the trellis. However, instead of using linear operations specified by generator polynomials to compute the encoder output for each branch, a nonlinear lookup-table directly assigns the encoder output values for each branch as in [2].

Denote the desired target sum-rate as R_{tgt}. The number n_0 of output bits per trellis branch is chosen, as a function of the number of users N, as follows:

$$n_0(N) = N/R_{\text{tgt}}, \tag{24}$$

where R_{tgt} is chosen so that n_0 is a natural number.

Denote the desired (optimal) ones density as p. In order to provide the required ones density, each of the branches needs to have the proper Hamming weights (W_h). Consider a 2^ν-state encoder producing a trellis with $B = 2^{\nu+1}$ branches in each stage. To achieve an optimal ones density of p, there should be B_w branches with Hamming weight $w = \lfloor p \cdot n_0 \rfloor$ and $B_{w+1} = B - B_w$ branches with Hamming weight $w+1$, where B_w should be chosen to minimize the deviation (Δ) from the desired ones density:

$$\Delta = |p \cdot n_0 - (B_{w+1} \cdot (w+1) + B_w \cdot w)/B|. \tag{25}$$

Given the Hamming weights of each branch, the remaining task is to assign the positions of the ones in each label. We provide a design technique that assigns these positions with the goal of maximizing the minimum distance between codewords. This design technique uses a non-standard definition of distance specific to the Z-Channel and the BAC. We introduce this distance in the next section.

A. Directional Hamming Distance, Z-Channel ML Decoding

Consider any two words of length n bits, $X = \{x_1, \cdots, x_n\}$ and $\tilde{X} = \{\tilde{x}_1, \cdots, \tilde{x}_n\}$. Define the Directional Hamming Distance $d_D(X, \tilde{X})$ as the number of positions where $x_i = 0$ and $\tilde{x}_i = 1$ for $i \in \{1, \cdots, n\}$. Note that $d_D(X, \tilde{X})$ is not necessarily equal to $d_D(\tilde{X}, X)$.

Denote the received word as $Y = \{y_1, \cdots, y_n\}$. Given the received word Y, any possible transmitted codeword X on the Z-Channel must satisfy $d_D(Y, X) = 0$, since there cannot be any one-to-zero transitions. The most likely transmitted codeword $\hat{X}$, is the codeword that minimizes the number of zero-to-one transitions $d_D(X, Y)$ among those codewords X satisfying $d_D(Y, X) = 0$. Hence, the ML decoder for the Z-Channel chooses the codeword $\hat{X}$ as:

$$\hat{X} = \text{argmin}_{X \in \mathcal{N}} \Big[d_D(X, Y) \Big], \tag{26}$$

where $\mathcal{N}$ is the set of codewords that satisfy $d_D(Y, X) = 0$.

As in (4), let α be the probability of a zero-to-one transition in the Z-Channel. Using Eq. (26), the directional pairwise error probability between two different codewords X and $\tilde{X}$ (the probability of transmitting X and decoding $\tilde{X}$ if those were the only two codewords in the codebook) under ML decoding is:

$$P_e(X \to \tilde{X}) = \begin{cases} \frac{1}{2}\alpha^{d_D(X,\tilde{X})} & W_H(X) = W_H(\tilde{X}) \\ \alpha^{d_D(X,\tilde{X})} & W_H(X) < W_H(\tilde{X}) \\ 0 & W_H(X) > W_H(\tilde{X}) \end{cases} \tag{27}$$

where $W_H(\cdot)$ denotes the Hamming weight. If two codewords have different Hamming weights, the codeword with the smaller Hamming weight will never be incorrectly decoded by a maximum likelihood (ML) decoder when the code with the larger Hamming weight is transmitted. On the other hand, if both codewords have the same Hamming weight, the directional Hamming distances are equal and errors can be made in either direction. In any case, the directional distance that affects pairwise error probability is the larger of the two. Thus, a proper definition of pair-wise distance for the Z-Channel is:

$$d_Z(X, \tilde{X}) = \max[d_D(X, \tilde{X}), d_D(\tilde{X}, X)]\,. \tag{28}$$

This metric for the Z-Channel is well known, appearing in [6] and [7] among other papers.

B. Conservative Branch-Wise Distance for the Z Channel

The definition of distance in (28) for the Z channel cannot be applied branch-wise, since it is impossible to tell from an individual branch which codeword will end up having more Hamming weight. For that reason, we will use a conservative definition of distance for our trellis code design, considering both directional distances. Namely, our definition of branch-wise distance between any two branches b_i and b_j is

$$d_{D,\text{min}} = \min[d_D(b_i, b_j), d_D(b_j, b_i)]. \tag{29}$$

Our code designs maximize this conservative branch-wise metric.

With this branch-wise metric, codewords with equal Hamming weights produce larger values of $d_{D,\text{min}}$ than codewords with different Hamming weights, so we will assign output values to the trellis branches with Hamming weights as similar as possible, preferably equal.

C. Conservative Branch-Wise Distance for the BAC

For the case of the BAC, with zero-to-one transition probability α and one-to-zero transition probability β, The probability of Y (of length n) being received given that X was transmitted is

$$P(Y|X) = \alpha^{d_D(X,Y)} + \beta^{d_D(Y,X)} + (1-\alpha)^{n - W_H(Y) - d_D(Y,X)} + (1-\beta)^{W_H(Y) - d_D(X,Y)}.$$

The ML decoder chooses $\hat{X}$ to maximizes $P(Y|X)$:

$$\hat{X} = \text{argmin}_X \Big[d_D(X, Y) \ln\Big(\frac{1-\beta}{\alpha}\Big) + d_D(Y, X) \ln\Big(\frac{1-\alpha}{\beta}\Big) \Big].$$

We have not found a simple relationship between the directional distances $d_D(X, \tilde{X})$ and $d_D(\tilde{X}, X)$, and the pairwise error probability $P_e(X \to \tilde{X})$. However, this work considers channels with significant asymmetry, where β is much smaller than α. In that case, the pairwise error probability may be approximated by the previous discussion. Thus, when designing the NLTC, the same conservative branch-wise definition of distance $d_{D,\text{min}}$ is used for both the Z-Channel and the BAC.

D. Nonlinear Trellis Code Design

The main task of our nonlinear trellis code (NLTC) design is to assign output values to the branches of the trellis to maintain the desired average ones density p while maximizing the minimum directional distance $d_{D,\text{min}}$ of the NLTC given the parameters of code (ν, p and n_0). We propose a design technique that provides a lower bound on the minimum conservative distance and maximizes it by extending Ungerboeck's rules for trellis design [10]. The design technique consists of the following steps.

The trellis paths of two valid codewords split from a common state at some trellis section, and merge to a common state at some other trellis section. Since a feed-forward encoder is used, two valid codewords must traverse different branches produced by a common input in ν consecutive trellis sections before a merge. The design procedure begins by ensuring all branches produced by the same input to have a conservative distance ($d_{D,\text{min}}$) of at least 1 between each other. Thus, if each of those sections adds at least 1 to the conservative distance, then for the overall trellis $d_{D,\text{min}} \geq \nu$. This can be accomplished if $\binom{n_0}{w} \geq 2^\nu$, where $w = \lfloor p \cdot n_0 \rfloor$. This last inequality is satisfied in the applications considered in this work, since the code-rates are very small (n_0 is large).

Once the weights of the branches are chosen, and branch labels are selected to ensure $d_{D,\text{min}} \geq 1$, we assign branch

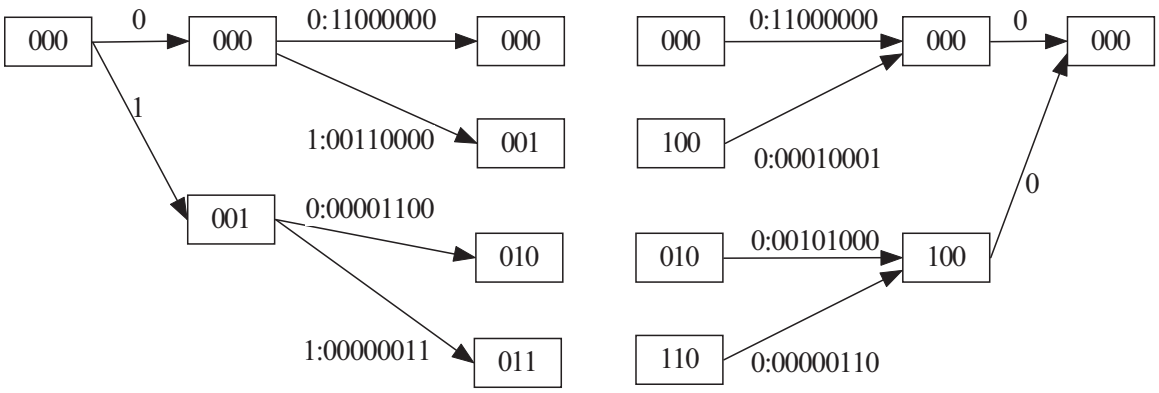


Fig. 4. Example of extension of Ungerboeck's distance bound focusing on splits and merges. Shown at left are the 4 branches produced by a split from the all-zero state in the previous trellis section. Shown at right are the 4 branches that merge to the all-zero state in the next trellis section. Branches are labeled (input:output). For both splitting and merging, any pair of branches in the four branches belonging to the group all have the maximum $d_{D,\min}$ of 2.

labels to the branches. Our approach for this assignment is based on Ungerboeck's idea of maximizing the distance between splits and merges [10]. Given that our branch labels will have Hamming weights of w or $w+1$, the best we can hope for is to guarantee that $d_{D,\min} \geq w$ between branches that share a split or merge.

Ungerboeck's rule can be extended more deeply into the trellis, and maximize not only the distance between splits, and the distance between merges, but the distance between the 4 branches emanating from a split in the previous trellis section, or the 8 branches emanating from a split two sections before, and so on. The same can be done with the merges moving backwards in the trellis.

Notice that lower bounding by w the distance for all sets of 8 branches emanating from a split two sections earlier also lower bounds by w the distance between all 4 branches emanating from any split a single trellis section earlier, and lower bounds by w every pair of branches at the beginning of any split. The same idea applies to the merges. If we consider h sections after a split, and g sections before a merge, the new bound for the minimum conservative distance is of the overall trellis is

$$d_{D,\min} \geq (w-1)(h+g) + \nu + 1. \tag{30}$$

The sum $h+g$ for which (30) can hold is limited by the parameters of the design. For example $h+g \leq \nu+1$. All the branches of the relevant splits and merges must have distance of at least w between each other when (30) is satisfied. Thus, the sum $h+g$ is limited by constraints resulting from the requirement that h and g must be small enough that the relevant sets of branches in each of the h or g trellis sections of the split or merge are all separated by the maximum conservative distance. Note that the condition need only be enforced for the trellis section involving the most branches (i.e. the last section of a split or the earliest section of a merge). The condition will then automatically be satisfied by the trellis sections involving fewer branches since these smaller groups are themselves strict subsets of larger groups that meet the enforced condition.

From the splitting point of view, the largest groups contain 2^h branches, which should have conservative distance of at least w between each other. Satisfying $w \cdot 2^h \leq n_0$ is required to guarantee a conservative distance of w between any two branches in a group of 2^h branches. From the merging point of view, the largest groups contain 2^g branches, and therefore the requirement is $w \cdot 2^g \leq n_0$. Note that each branch belongs to one group of 2^h and one group of 2^g, but no pair of branches belongs to the same two groups.

As an example, consider a rate-1/8 ($n_0 = 8$) 8-state trellis ($\nu = 3$), where a ones density of $p = 1/4$ is required. The Hamming weight of each output must be $w = n_0 \cdot p = 2$. There are $2^{\nu+1} = 16$ branches. There are $\binom{8}{2} = 28$ possible outputs with $w = 2$. Hence, we can choose 16 different outputs (from the 28 possibilities) with $d_{D,\min} \geq 1$ between each other.

Since $w = 2$, the maximum conservative distance between two outputs is 2. The maximum number of outputs with $d_{D,\min} = 2$ between each other is $n_0/w = 4$. Therefore we can choose $h = g = 2$ so that $2^h = 2^g = 4$.

For an eight-state code, $h = 2$ implies that we need to maximize the distance between every group of branches departing from the states (abX), with ab fixed for each group and $X \in \{0,1\}$, with any input (there are 4 branches in each group). To satisfy $g = 2$ we maximize the distance between any group of branches departing from states (XXc) with the same input (there are also 4 in each group).

Fig. 4 shows an example for $a = 0$, $b = 0$, and $c = 0$. Specifically, it shows the 4 branches produced by a split from the all-zero state in the previous trellis section (at left) and the 4 branches that can produce a merge to the all-zero state in the next trellis section (at right). In this example, it is the branch connecting the zero state to itself that appears in both the splitting group and the merging group. For both splitting and merging, any pair of branches in the four branches belonging to the group has the maximum $d_{D,\min}$ of 2.

Table I shows a labeling that achieves (30) for the example of a rate-1/8 ($n_0 = 8$) 8-state trellis ($\nu = 3$), with a ones density of $p = 1/4$ and $h = g = 2$. Table I is constructed so that outputs in the same row (corresponding to a $g = 2$ merge) or the same column (corresponding to an $h = 2$ split) do indeed have $d_{D,\min} = 2$. Therefore, using (30) the minimum distance of the code is $d_{D,\min} = 8$, which is the maximum possible conservative distance given the parameters of the example. The first column of Table I is illustrated in

TABLE I

EXAMPLE OF LABELING DESIGN. S DENOTES THE CURRENT STATE, u DENOTES THE INPUT BIT. EACH OUTPUT CORRESPONDS TO THE BRANCH PRODUCED BY THE INPUT u WHEN THE CURRENT STATE IS S.

S	u	output	S	u	output	S	u	output	S	u	output
000	0	11000000	010	0	00101000	100	0	00010001	110	0	00000110
000	1	00110000	010	1	10000010	100	1	01000100	110	1	00001001
001	0	00001100	011	0	01000001	101	0	00100010	111	0	10010000
001	1	00000011	011	1	00010100	101	1	10001000	111	1	01100000

the left portion of Fig. 4. The first row of Table I is illustrated in the right portion of Fig. 4.

IV. HANDLING A LARGE NUMBER OF USERS

For a target sum-rate R_{tgt} and a specified target BER, there may be a limitation on the number of users N using the NLTCs of this paper if the number of states 2^{ν} is not large enough. As seen in Section II-A, the optimal ones density for a certain number of users is well approximated by (17), where δ depends on the channel (see Fig. 3).

Let $W_b(N)$ be the total number of ones in all of the $2^{\nu+1}$ branches in a single trellis section of a code designed for N users. We have the following:

$$W_b(N) \approx 2^{v+1} n_0 p(N) \tag{31}$$

$$= \frac{N(1-\delta^{1/N})2^{v+1}}{R_{\text{tgt}}}, \tag{32}$$

where (32) follows from (24) and (17). $W_b(N)$ increases monotonically with the number of users N. Using (23) we have

$$W_b(N) \to \frac{-\ln(\delta)2^{v+1}}{R_{\text{tgt}}}, \tag{33}$$

so that $W_b(N)$ converges to a finite limit as N becomes large.

On the other hand, from Eq. (24) the number $n_0(N)$ of output bits per trellis section linearly increases with N. Hence, for a large enough number of users, $n_0(N)$ becomes greater than $W_b(N)$. Let N_c denote the smallest number of users at which $n_0(N_c) \geq W_b(N_c)$.

The design of a code for N_c users is straightforward. For each branch, add ones in positions that aren't used in previous branches until its assigned Hamming weight is reached. Moreover, the best code for N_c users is essentially the best code for any number of users greater than N_c. The only difference is that as N grows more zeros are added to the output.

The channel degrades as N increases, hence degrading the code performance. However, for N_c sufficiently large, this degradation becomes marginal, as both α and β converge to fixed values. For example, in the OR-MAC, using (10) in (11) yields:

$$\alpha(N) = 1 - (1/2)^{(N-1)/N}, \tag{34}$$

which converges to 1/2 as N goes to ∞. N_c increases with ν, so choosing ν sufficiently large (e.g. large enough to handle the $\alpha = 1/2$ case on the OR-MAC), a target sum rate can be achieved regardless of the number of users with essentially the same performance.

Fig. 5 shows the number of output bits per trellis section n_0 and the total number of ones W_b in all the branches of a trellis section vs. the number of users, for $\nu = 5$ and $\nu = 6$ codes designed for the OR-MAC, using a target sum rate of $R_{\text{tgt}} = 0.3$ and $\delta = 1/2$. With $\nu = 5$, $N_c = 44$ and $\alpha(N_c) = 0.492$. With $\nu = 6$, $N_c = 89$ and $\alpha(N_c) = 0.496$, which is already very close to $\alpha(\infty) = 0.5$.

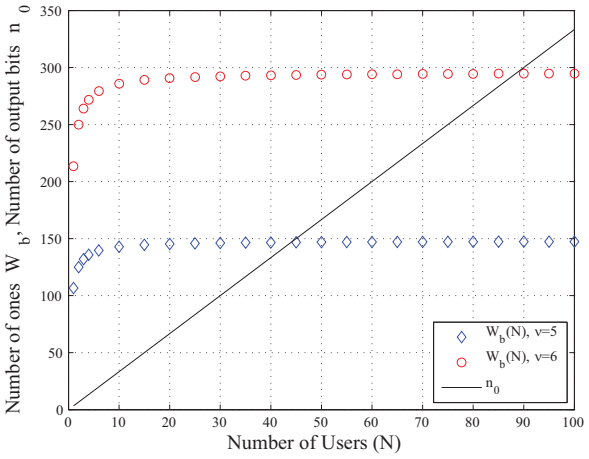


Fig. 5. Total number of ones (W_b) in the output of all branches of a single trellis section and number of output bits per branch (n_0) vs. number of users (N), for the OR-MAC with $R_{\text{tgt}} = 0.3$ for $\nu = 5$ and $\nu = 6$. For the OR-MAC, $\delta = 1/2$.

The question is whether a code designed for $N_c = 44$ can continue to perform well as the number of users increases beyond 44 (with proper added zeros) and α increases beyond 0.492 towards 1/2. If not, can the $\nu = 6$ code designed for $N_c = 89$ continue to perform well as the number of users increases beyond 89 (with the proper added zeros to the output) and α increases beyond 0.492 towards 1/2? As will be corroborated in Section VI, $\nu = 6$ is sufficient to achieve the target sum-rate of 0.3 with consistent performance regardless of the number of users.

V. TRANSFER FUNCTION BOUND FOR NLTC CODES

Ellingsen [11] provided a combinatorial expression for an upper bound on the BER of linear block codes over the Z-channel under ML decoding. For convolutional codes assuming binary PAM or QPSK, Viterbi [12] introduced an analytical technique using generating functions to provide a union bound on the BER of convolutional codes. Viterbi's technique is based on a 2^{ν}-state diagram for the convolutional encoder. In the case of general trellis codes where high level constellations introduce nonlinearity, Biglieri [13][14] generalized Viterbi's algorithm by using the product state diagram with $2^{2\nu}$-states. Biglieri's algorithm can be applied to nonlinear trellis codes over the Z-channel, and more generally over the BAC, with modifications on the pairwise error probability measure.

A. Transfer Function Bound Over the Z-Channel

The directional pairwise error probability between two different codewords X and $\tilde{X}$ under ML decoding over the Z-Channel is shown in (27). Hence, for a specified pair of

codewords X and $\tilde{X}$, the sum of the two directional pairwise error probabilities is

$$P_e(X \to \hat{X}) + P_e(\hat{X} \to X) \tag{35}$$
$$= \alpha^{\max(d_D(X,\hat{X}),d_D(\hat{X},X))} \tag{36}$$
$$\leq \frac{1}{2}\left(\alpha^{d_D(X,\hat{X})} + \alpha^{d_D(\hat{X},X)}\right) \tag{37}$$

Thus, if $P_e(X \to \hat{X})$ is replaced (not always upper-bounded) by $\frac{1}{2}\alpha^{d_D(X,\hat{X})}$ for all the codeword pairs X and $\hat{X}$, the transfer function bound technique applied to the NLTC will still yield a valid overall upper bound because of (37).

As in [13], the product state diagram consists of state pairs, (s_e, s_r), where s_e is the encoder state and s_r the receiver state. Following Biglieri's notation, the product states can be divided into two sets, the good states denoted by S_G and the bad states denoted by S_B, defined respectively as follows:

$$S_G = \{(s_e, s_r) \mid s_e = s_r\},\ S_B = \{(s_e, s_r) \mid s_e \neq s_r\}. \tag{38}$$

By suitably renumbering the product state pairs with single-value states s, we get the transition matrix

$$S(W,I) = \left[\begin{array}{c|c} S_{GG}(W,I) & S_{GB}(W,I) \\ \hline S_{BG}(W,I) & S_{BB}(W,I) \end{array}\right], \tag{39}$$

where the $N_s \times N_s$ matrix $S_{GG}(W,I)$ accounts for transitions between two good product states, the $N_s \times (N_s^2 - N_s)$ matrix $S_{GB}(W,I)$ accounts for transitions from good product states to bad product states, and so forth. $N_s = 2^\nu$ is the number of encoder states.

The (i,j) entry of $S(W,I)$ is a branch label for the transition from state $s = i$ to state $s = j$ in the product state diagram. The $i \to j$ branch is labeled by

$$p(i \to j) W^{d_D(x_e,x_r)} I^{d_H(u_e,u_r)}, \tag{40}$$

where $d_H(\cdot,\cdot)$ denotes the Hamming distance, u_e and x_e denote the input and output word for the encoder states respectively, and u_r and x_r denote the input and output word for the receiver states. The probabilities $p(i \to j)$ are $1/2$ for rate-$1/n$ codes (more generally 2^{-k} for rate-k/n codes).

The transfer function $T(W,I)$ is

$$T(W,I) = p_s\{S_{GG} S_{GB}(I - S_{BB})^{-1} S_{BG}\}\mathbf{1}, \tag{41}$$

where $p_s = [\frac{1}{N_s}\frac{1}{N_s}\cdots\frac{1}{N_s}]$ is the marginal probability distribution of the encoder states and $\mathbf{1} = [11\cdots1]^T$. The BER bound is computed as

$$BER \leq \frac{1}{2}\cdot\frac{1}{k}\cdot\left.\frac{\partial T(W,I)}{\partial I}\right|_{W=\alpha,I=1}. \tag{42}$$

B. Transfer Function Bound Over the BAC

For the BAC, using a variation of the Bhattacharyya bounding technique [15], the sum of the error probabilities of transmitting either sequence and decoding the other can be upper bounded by:

$$P\left(X \to \hat{X}\right) + P\left(\hat{X} \to X\right) \tag{43}$$
$$= \sum_Y \min\left\{P\left(Y|X\right), P\left(Y|\hat{X}\right)\right\} \tag{44}$$
$$\leq \sum_Y \sqrt{P\left(Y|\hat{X}\right) P\left(Y|X\right)} \tag{45}$$
$$= \prod_i \sum_{y_i} \sqrt{P\left(y_i|\hat{x}_i\right) P\left(y_i|x_i\right)} \tag{46}$$
$$= \prod_i g(i) \tag{47}$$
$$= \left(\sqrt{\alpha(1-\beta)} + \sqrt{\beta(1-\alpha)}\right)^{d_H\left(X,\hat{X}\right)}, \tag{48}$$

where

$$g(i) = \begin{cases} 1 & x_i = \hat{x}_i \\ \sqrt{\alpha(1-\beta)} + \sqrt{(1-\alpha)\beta} & x_i \neq \hat{x}_i \end{cases}. \tag{49}$$

Replacing the branch label of Eq. (40) by

$$p(i \to j) W^{d_H(x_e,x_r)} I^{d_H(u_e,u_r)}, \tag{50}$$

the BER is upper bounded by

$$BER \leq \frac{1}{2}\cdot\frac{1}{k}\cdot\left.\frac{\partial T(W,I)}{\partial I}\right|_{W=\sqrt{\alpha(1-\beta)}+\sqrt{\beta(1-\alpha)},I=1}. \tag{51}$$

VI. Performance Results

We have tested NLTC performance over the uncoordinated OR-MAC and CI-MAC with single-user decoding for different numbers of users varying from 6 to 1500.

A. NLTC on the OR-MAC

Fig. 6 shows the BER (obtained by C++ simulation) of three 64-state NLTC codes designed to work in a 6-user OR-MAC, along with their transfer function bounds computed using (42). The codes are a rate-1/17 NLTC code with $p = 2/17$, a rate-1/18 NLTC code with $p = 1/8$ and a rate-1/20 NLTC code with $p = 1/8$. The exact specification of all the codes presented in this paper can be found in [16].

The ones densities of $p = 1/8$ and $p = 2/17$ are close to 0.1079, the optimal density for a 6-user OR-MAC with single-user decoding (using (17) with $\delta = 0.5040$ which maximizes (8) for six users). Dark circles show the BER at the values of α produced by the 6-user OR-MAC with single-user decoding. The sum-rates achieved are $R_6^{\text{SUD}} = 6/17 \approx 0.353$, $R_6^{\text{SUD}} = 1/3 \approx 0.333$ and $R_6^{\text{SUD}} = 0.3$ respectively. Note that these sum rates are significantly lower that the theoretical sum rate of $\ln 2 = 0.6931$ just as trellis codes fall short of approaching capacity on the AWGN channel. In [5] we demonstrate nonlinear turbo codes that achieve a sum rate of 0.6 on the OR-MAC with BER below 10^{-6} for up to 24 users.

The 3 sum-rate points discussed above have been simulated both with six users employing IDMA with single-user decoding and with a single user communicating over the Z-channel with the corresponding value of α . Both cases gave the same bit error rates (within the expected variation due to Monte Carlo simulation with 100 errors), which corroborates the theory.

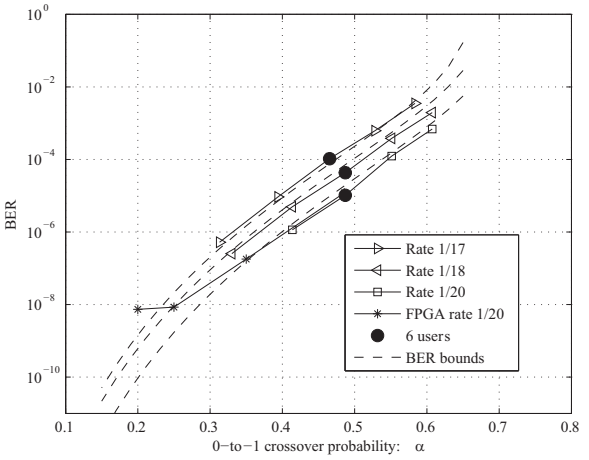


Fig. 6. Bit Error Rate (BER) of three 64-state NLTC codes versus the Z-Channel crossover probability (α). Dashed lines show the BER bounds for each rate using (42). For all three rates C++ simulation results are shown. Additionally, FPGA simulation results are shown for the rate-1/20 code. Dark circles show the BER at the values of α produced by the 6-user OR-MAC with single-user decoding.

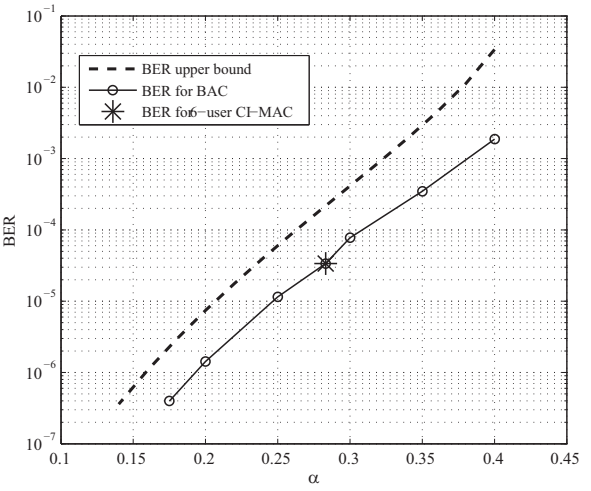


Fig. 7. BER of 64-state NLTC codes vs. α over the BAC. The asterisk shows the code performance over the 6-user CI-MAC.

TABLE II

PERFORMANCE OF 64-STATE NLTCS FOR DIFFERENT NUMBERS OF USERS (N) AND A SUM-RATE $R_N^{\text{SUD}} = 0.3$, ON THE OR-MAC.

N	n_0	R_N	α	BER
6	20	0.3	0.439	1.0214×10^{-5}
100	344	0.291	0.4777	1.1046×10^{-5}
300	1000	0.3	0.4901	1.2157×10^{-5}
900	3000	0.3	0.4906	1.2403×10^{-5}
1500	5000	0.3	0.4907	1.2508×10^{-5}

The transfer function bounds are very close to the simulated performance in all three cases. The transfer function bound is an upper bound on the expectation assuming maximum likelihood decoding, and it is not unexpected for a finite-traceback-depth simulation to be slightly above the bound (as in the case of the rate-1/17 code). Our decoding depth is 35 for this simulation. Also, some variation around the expectation is to be expected.

In order to prove that NLTC codes are feasible today for high speeds, a hardware demonstration was built using fiber optics and Xilinx Virtex2-Pro 2V20 FPGAs. The implementation had an equivalent gate count of 360K gates and is able to encode and decode the rate-1/20 NLTC code concatenated with a Reed-Solomon block code at an information rate of 70Mbps. A detailed description of the FPGA implementation can be found in [17], [18].

Results for the rate-1/20 NLTC code obtained in the FPGA testbed are also shown in Fig. 6. Due to design constraints, the hardware Viterbi decoder has a maximum path distance metric of 20, and hence 1-to-0 transitions are given a distance of 20 instead of ∞. This difference causes the deviation from the theoretical bound at low bit error rates.

Section IV suggested that for the OR-MAC, a 64-state NLTC has enough states to perform well for even a large number of users. Table II shows BERs for 6 to 1500 users. The performance is practically the same for all the cases corroborating the analysis of Section IV.

B. Concatenation of NLTC Code With a Block Code

A good solution for applications that require a very low BER is to include as an outer code a high-rate block code that can correct a small number of symbol errors, dramatically lowering the BER.

We implemented concatenation of the rate-1/20 NLTC code with a (255-byte, 247-byte) Reed-Solomon code for the 6-user OR-MAC scenario. The rate of this code is (247/255)(1/20) = 0.0484 which gives a sum-rate of approximately 0.2906. The observed BER was 2.48×10^{-10}. Although simulations with the concatenated Reed-Solomon code for more than 6 users have not been performed, it can be inferred from results of Section VI-A that the system proposed in this work can achieve almost 30% of full capacity, with a BER on the order of 10^{-10} even for a large number of users.

C. NLTC for 6-User Coherent Interference MAC

Fig. 7 shows the performance of a 64-state NLTC code on the CI-MAC with single-user decoding. This rate-1/30 NLTC gives a sum-rate of 0.2 for six users. It has a ones density $p = 1/15$. It was designed for the 6-user CI-MAC, which with single-user decoding has a maximum theoretical sum-rate of $R_6^{SUD} = 0.48$ with a ones density of $p = 0.059$. Fig. 7 also shows the analytical bound computed using (51). This analytical bound is not as tight as for the Z-Channel since (48) is not as tight as (37), which was used for the Z channel.

Large numbers of users can also be handled on the CI-MAC. However, more NLTC states are required than for the OR-MAC. Table III shows the performance of 128-state NLTC codes for a sum-rate of 0.2, and different numbers of users.

VII. CONCLUSIONS

This paper considers binary-input binary-output multiple access channels (MACs), focusing on the OR-MAC in which users' binary transmissions are combined using the logical OR

TABLE III
PERFORMANCE OF 128-STATE NLTC FOR DIFFERENT NUMBERS OF USERS N AND A SUM-RATE $R_N = 0.2$, ON THE CI-MAC.

N	R_N	α	β	BER
6	0.2	0.2832	0.0622	1.46×10^{-5}
32	0.2	0.3107	0.0664	2.71×10^{-5}
104	0.2	0.3147	0.0677	6.35×10^{-5}

operator. With the OR-MAC, the sum of all the user rates can be 1 with coordination such as time division or joint decoding, but this paper is interested in the uncoordinated scenario in which there is no time division and each receiver employs single-user decoding (SUD).

Specifically, this paper designs trellis codes for use in an interleave(r)-division multiple access (IDMA) scheme [3] to provide uncoordinated use of the OR-MAC by numerous users. Good performance in this scenario hinges on each user transmitting with a ones density that is significantly less than 50%. The needed low ones densities require nonlinear trellis codes. This paper provides design metrics and design techniques that deliver the needed trellis codes both for the OR-MAC and for a more general OR-with-Interference (ORI) MAC. Specific code designs and simulation results are provided.

A new transfer function union bound provides extremely accurate BER performance for the new nonlinear trellis codes on the Z-channel induced by the OR-MAC. A new transfer function union bound using the Bhattacharyya bounding technique is also provided for the general binary asymmetric channels such as those induced by the ORI-MAC. Though not as accurate as the Z-channel characterization, this bound still gives a good BER characterization for the ORI-MAC.

Under SUD, the highest possible sum rate for the OR-MAC is $\ln 2$ which is about 70% of the sum rate with time division. The new nonlinear trellis codes can provide 30% of the time-division OR-MAC sum rate while delivering a BER on the order of 10^{-5}. The concatenation of these codes with a high-rate Reed Solomon achieves 29% of the time-division OR-MAC sum rate while delivering a BER on the order of 10^{-10}. In other work [5] we have designed nonlinear turbo codes that achieve 60% of the time-division OR-MAC sum rate while delivering a BER on the order of 10^{-6}.

Still, the nonlinear trellis codes of this paper provide the option of sacrificing a portion of sum rate in order to permit a system to be completely uncoordinated and to use low-complexity Viterbi decoding. This uncoordinated approach is especially attractive in environments with a large number of users, and we provided simulation results showing that the IDMA-NLTC approach can support as many as 1500 users on the OR-MAC while still achieving 30% of the time-division OR-MAC sum rate and delivering a BER on the order of 10^{-5}.

The OR-MAC can be used as a simple communication model that describes the multiple-user local area network optical channel with non-coherent combining. We have built a 6-user optical system transmitting data on a single wavelength to demonstrate the IDMA-NLTC approach in an optical setting. This system employs fiber optics and Xilinx Virtex2-Pro 2V20 FPGAs. The implementation has an equivalent gate count of 360K gates and is able to encode and decode the rate-1/20 NLTC code concatenated with a Reed-Solomon block code at an information rate of 70Mbps. A detailed description of the FPGA implementation can be found in [17], [18].

REFERENCES

[1] A. J. Grant, B. Rimoldi, R. L. Urbanke, and P. A. Whiting, "Rate-splitting multiple access for discrete memoryless channels," *IEEE Trans. Inf. Theory*, vol. 47, pp. 873–890, Mar 2001.

[2] P. Chevillat, "N-user trellis coding for a class of multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 27, Jan. 1981.

[3] L. Ping, L. Liu, K. Wu, and W. K. Leung, "Interleave division multiple-access" *IEEE Trans. Wireless Commun.*, vol. 5, pp. 938–947, Apr. 2006.

[4] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Wiley Series in Telecommunications, 2006.

[5] M. Griot, A. I. Vila Casado, and R. D. Wesel, "Non-linear turbo codes for interleaver-division multiple access on the OR channel," in *Proc. IEEE Global Telecommun. Conf.*, Nov.–Dec. 2006.

[6] T. Kløve, "Error correcting codes for the asymmetric channel," Dept. Mathematics, Univ. Bergen, Bergen, Norway, Tech. Rep. 18-09-07-81, 1995.

[7] F.-W. Fu, S. Ling, and C. Xing, "New lower bounds and constructions for binary codes correcting asymmetric errors," *IEEE Trans. Inf. Theory*, vol. 49, pp. 3294–3299, Dec. 2003.

[8] A. Bennatan and D. Burshtein, "On the application of LDPC codes to arbitrary discrete-memoryless channels," *IEEE Trans. Inf. Theory*, vol. 50, pp. 417–438, Mar. 2004.

[9] E. A. Ratzer and D. J. C. MacKay, "Sparse low-density parity-check codes for channels with cross-talk," in *Proc. Inf. Theory Workshop*, Apr. 2003, pp. 127–130.

[10] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inf. Theory*, vol. 28, pp. 55–67, Jan. 1982.

[11] P. Ellingsen, S. Spinsante, and O. Ytrehus, "Maximum likelihood decoding of codes on the asymmetric Z-channel," in *Proc. 10th IMA Int. Conf. Cryptography Coding*, 2005.

[12] A. Viterbi, "Convolutional codes and their performance in communication systems," *IEEE Trans. Commun.*, vol. 19, pp. 751–772, Oct 1971.

[13] E. Biglieri, "High-level modulation and coding for nonlinear satellite channels," *IEEE Trans. Commun.*, vol. 32, pp. 616–626, May 1984.

[14] Y.-J. Liu, I. Oka, and E. Biglieri, "Error probability for digital transmission over nonlinear channels with application to TCM," *IEEE Trans. Inf. Theory*, vol. 36, pp. 1101–1110, Sep. 1990.

[15] M. Griot, W.-Y. Weng, and Richard D. Wesel, "A tighter Bhattacharyya bound for decoding error probability," *IEEE Commun. Lett.*, Apr. 2007.

[16] [Online]. Available: http://www.ee.ucla.edu/∼csl/files/codes/nltc.html

[17] H. Chan, M. Griot, A. Vila Casado, R. D. Wesel, and I. Verbauwhede, "High speed channel coding architectures for the uncoordinated OR channel," in *Proc. IEEE 17th Int. Conf. Application-Specific Systems, Architectures Processors (ASAP)*, Sep. 2006.

[18] H. Chan, A. Vila Casado, J. Basak, M. Griot, W.-Y. Weng, R. D. Wesel, B. Jalali, E. Yablanovitch, and I. Verbauwhede, "Demonstration of uncoordinated multiple access in optical communications," *IEEE Trans. Circuits Syst. I*, vol. 55, no. 10, pp. 3259–3269, Nov. 2008.

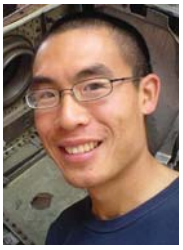
[19] M. Griot, A. I. Vila Casado, W.-Y. Weng, H. Chan, J. Basak, E. Yablanovitch, I Verbauwhede, B. Jalali, and R. D. Wesel, "Trellis codes with low ones density for the OR multiple access channel," in *Proc. IEEE International Symp. Inf. Theory*, July 2006.

[20] M. Griot, A. I. Vila Casado, W.-Y. Weng, H. Chan, J. Basak, E. Yablanovitch, I. Verbauwhede, B. Jalali, and R. D. Wesel, "Interleaver-division multiple access on the OR channel," in *Proc. Workshop Inf. Theory Applications (ITA)*, Feb. 2006.

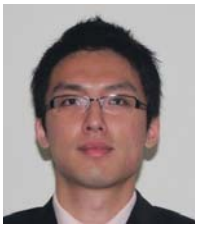
Miguel Griot received his B.S. degree from the Universidad de la Republica, Uruguay, in 2003, and his M.S. and Ph.D. degrees from the University of California at Los Angeles, in 2004 and 2008, respectively, all in electrical engineering. He is currently with Qualcomm Inc., San Diego, CA. His research interests include wireless communications, channel coding, information theory, multiple access channels, and broadcast channels.

Andres I. Vila Casado received his B. S. in electrical engineering from Politecnico di Torino, Turin, Italy in 2002. He received his M. S. and Ph. D. degrees in electrical engineering from the University of California, Los Angeles, in 2004 and 2007, respectively. At UCLA and at Politecnico di Torino he conducted research on communication theory with a focus on channel coding and information theory. He is currently a Research Scientist at Mojix, Inc. where he conducts research on physical layer communications and Bayesian estimation for RFID applications.

Wen-Yen Weng received his B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan in 1997. He received his M.S. and Ph.D. in electrical engineering from the University of California, Los Angeles, in 2001 and 2007, respectively. He was an Assistant Professor in the Computer Science and Information Engineering Department at Chung-Hua University, Hsin-Chu, Taiwan from 2007 to 2010. He joined Ralink Technology in 2010 as a Deputy Technical Manager, focusing on Wi-Fi system algorithm development. His research interests include channel coding and signal processing for communication systems.



Herwin Chan received the B.A.Sc. degree from the University of British Columbia, Vancouver, BC, Canada, and the M.S. and Ph.D. degrees in electrical engineering from the University of California, Los Angeles. He is currently the embedded systems architect at Pharmaco-kinesis Corporation. His research interests include security and hardware/software co-design of embedded systems.



Jiadong Wang received his B.S. degree in automation from Tsinghua University, Beijing, China, in 2007 and the M.S. degree in electrical engineering from UCLA in 2008. He is currently pursuing his Ph.D. in electrical engineering at UCLA. Since 2007, he has been a research assistant at the Communication Systems Laboratory (CSL) under the mentorship of Professor Richard D. Wesel. His research is in the area of communication theory with a focus on channel coding. He has investigated topics including LDPC codes and turbo codes for applications such as flash memory and broadcast channels.

Richard D. Wesel is a Professor with the UCLA Electrical Engineering Department and is the Associate Dean for Academic and Student Affairs for the UCLA Henry Samueli School of Engineering and Applied Science. He joined UCLA in 1996 after receiving his Ph.D. in electrical engineering from Stanford. His B.S. and M.S. degrees in electrical engineering are from MIT. His research is in the area of communication theory with particular interest in channel coding. He has received the National Science Foundation CAREER Award, an Okawa Foundation award for research in information and telecommunications, and the Excellence in Teaching Award from the Henry Samueli School of Engineering and Applied Science. He has authored or co-authored over 130 conference and journal publications.