On Quasi-Cyclic Repeat-Accumulate Codes

R. Michael Tanner

Department of Computer Science
University of California, Santa Cruz
Santa Cruz, CA 95064
tanner@cse.ucsc.edu

Abstract

Repeat-Accumulate (RA) codes, consisting of a simple binary repetition followed by a permutation and an accumulator, are converting into quasi-cyclic codes with multiple cross-linked cycles. The algebraic heuristics are used to design codes with an explicit algebraic permutation yielding desirable graph properties. Recursive graph constructions techniques are given for the construction of very long codes. Simulation results for codes of rate 1/3 and 1/4 demonstrate that these algebraic RA codes can compete with RA codes using pseudo-random permutations and outperform the random codes at high SNR.

1 Introduction

Divsalar, Jin, and McEliece [3] define a class of "turbo-like" codes [2] called *repeat and accumulate* (RA) codes in which an input block of N bits is repeated q times, then passed through a permutation or scrambler, and then input to a rate 1 two-state convolutional encoder, a simple accumulator. In some sense, RA codes are the simplest possible serial concatenated code of the turbo code style: neither the repetition nor the trivial convolutional code by itself achieves any coding gain. Nonetheless, the authors demonstrate that for sufficiently long permutations, RA codes perform remarkably well. By calculating the explicit input-output weight enumerator for these codes, they are able to show that for the ensemble of codes with randomly chosen permutations, the probability of block error for maximum likelihood decoding approaches zero as the block length increases to infinity at sufficiently high E_b/N_0. As the repetition factor q increases and the code rate decreases, the gap between their bounds and the Shannon limit decreases monotonically. The theoretical results for RA codes are supported by empirical studies showing that suboptimal iterative decoders also perform surprisingly well. Indeed, for code rates of $1/q, q \geq 3$, RA codes are an interesting option for a very low complexity system.

For a given rate and block length, an RA code system is defined entirely by the permutation employed. While a random permutation on average is good, any specific permutation may engender some number of relatively low weight codewords. The presence of low weight words precludes very low output bit error rates (BER) at high signal-to-noise ratios, since the low weight words dominate the sum in union bounds on the bit error rate [1]. Random permutations are also problematic due to the memory required to store the addresses to which each input bit is mapped. It would be preferable to have a compact algorithm to generate the addresses, and to be able to prove that the

block code's minimum distance is large enough to not pose a significant source of error at higher E_b/N_0.

In this paper we give an algebraic prescription for a rate $1/q$ RA code of block length $N = qK$ when $qK = p(p-1)$ for some prime p such that q divides $p-1$. These codes do have a compact description and are amenable to algebraic analysis of the code space. We have not yet been able to devise a general bound on minimum distance, but simulations studies of two examples indicate that the minimum distance is superior to that of random codes. The program is as follows: A length N RA code is fashioned into a quasi-cyclic repeat-accumulate code (QCRA): first by tailbiting, requiring that the starting state and the ending state of the accumulator be the same, not necessarily zero; second, by imposing a regular periodicity on the permutation connections. The algebraically determined periodicity gives the code constraint (Tanner) graph a useful automorphism group. Effectively, the QCRA code is an algebraic time-varying tailbiting convolutional code, or equivalently, a quasi-cyclic code, with a very restricted form. To define long block length QCRA codes, a constraint graph is described recursively starting from a small graph with suitable valencies using what we will call a *quasi-cyclic extension*, or an *n-cyclic extension*, to make evident the length of the shift invariant cycles. This procedure yields a quasi-cyclic graph whose adjacency matrix is amenable to Fourier transform analysis. The recursive construction gives insights into the problem of constructing a code graph with comparatively few short cycle, assuring that the initial iterations of an iterative decoder will yield statistically independent estimates for a decoded bit. For some short codes, it is possible to prove a bound on minimum distance algebraically or computationally, exploiting the automorphism group. The codes of our prescription must have a minimum distance less than $q\sqrt{N}$. Simulations results for two algebraic QCRA codes show that they perform better than two random permutation competitors at high E_b/N_0, indicating that the minimum distance is almost certainly larger.

2 RA codes, the accumulator chain, and low weight words

Figure 1 gives a circuit diagram for a rate $1/q$ binary RA code and the associated code constraint graph. The information bits are of degree q (shown with q=3), the repetition factor. Codeword bits sent to the channel are shown as bit nodes of degree 2. While in a systematic version the information bits themselves are sent, we will assume that only the channel bits are transmitted. The simplicity of the rate 1 accumulator code is manifest in the constraint graph. The recursive equation of the accumulator imposes a parity check equation involving three variables: the current output channel bit value, the next input information bit value, and the next output channel bit value. An RA code constraint graph therefore always consists of a sequence of channel bits interspersed in alternation between parity nodes of degree 3, with the exact connection of the degree q information bit nodes to the parity nodes determined by the permutation used. The connection to a node will be referred to as a "tap" in the accumulator chain. Each information bit has q taps into the accumulator chain. After all the permuted information bits have entered the accumulator, it may not be in the zero state, and there is a practical question of how to terminate the transmission. One strategy is simply to allow some finite tail, repeating the final output value, and then to stop. This can induce minor end effects, where information bits with taps near the end of the transmission are more or less susceptible to error.

Arbitrarily viewing the time axis as moving from left to right, when the accumulator

is started in the zero state, the output channel bits will remain zero until the first non-zero bit causes the accumulator output to turn "on" and start emitting ones. It will continue to output ones until the next information bit causes it to "off" and return to the zero state. An RA codes with $q = 2$ will have low minimum distance because any cycle in the graph forms a codeword, and is easy to show that the minimum cycle can be no longer than $O(logN)$. For $q \geq 3$ a low weight word can arise from the close juxtaposition of all the taps for some small set of information bits. The extreme case, for example, with $q = 4$, is when the permutation chosen places two of an information bit's connection on successive parities in the chain and the other two taps on two other successive parity checks, resulting transmitted codeword of weight 2. More generally, if all the taps for a small set of bits lie in a small number of short zones of the accumulator, a low weight channel word can result. A first order test for the quality of a permutation is to check that the information bits that have taps in some sliding window of small size have their other taps widely dispersed elsewhere. Such a test is not sufficient to establish a minimum distance bound, however, because it will not necessarily detect the presence of some small set of non-zero information bits having taps clustered together in a few narrow zones.

A computational search for the minimum distance of an RA code can take advantage of a simple graph argument. For t non-zero information bits, there are qt ones entered into the accumulator chain, imply qt parities are active equations, and each non-zero channel bit can satisfy at most two of these parities. The codeword weight therefore must satisfy *Codeword Weight* $\geq q/2$ *(Input Weight).* The minimum weight word must arise from a low weight input, and for short codes one can exhaustively check all low weight information sequences. Obviously, an exhaustive examination of all low weight input words becomes infeasible as the code length increases.

3 RA Codes in Quasi-Cyclic Form

A cyclic code of length N is one for which the code automorphism group contains a cyclic group of order N that is one-transitive on the bits, allowing the bits to be organized into a single cycle of length N. Any cyclic shift *mod* N preserves the codespace. A *quasi-cyclic code* of length N is one for which the code is preserved by shifts of m positions, $m \geq 2$, for m a divisor of N ([6],pp. 256-261). In effect, this decomposes the code into m orbits of n bits, $n = N/m$, and the code is preserved under a cyclic shift *mod* n simultaneously in all m orbits.

The first step in forming a QCRA is to introduce "tailbiting" in the convolutional accumulator code. Instead of view the RA code as the output of a potentially semi-infinite accumulator chain, a chain containing N channel bits and N parity nodes can be closed on itself, forming a loop, with the last channel bit considered to be connected to the first parity node. The start state and the end state of the accumulator are forced to be the same. This is shown for $q = 3,\ N = 15$ QCRA code in Figure 2. For the sake of encoding and decoding, this might seem a nuisance, but it is of little consequence. For q even, there is always a codeword solution for any single non-zero information bit starting and ending in the zero state. The initial state is thus a free variable that can be viewed as an additional information bit. To eliminate the memory requirement, the initial and final state can be set to zero. For q odd, there is no solution for a single non-zero information bit. For the tailbiting RA code to have a solution, the parity of the incoming information bits must be even, guaranteeing that an even number of taps into

the accumulator chain are non-zero. By sacrificing one bit, no matter how large K, the initial and final state again becomes a free variable that can be set to zero. With this proviso, the minimum distance of the tailbiting RA code is at least as large as that of the initial RA code inasmuch as any word in the tailbiting RA code must be a word in the original RA code.

A tailbiting RA code is a quasi-cyclic code if the permutation of the inputs exhibits a shift invariance. Specifically, let m be a divisor of N, $mn = N$ and index the parity nodes in the accumulator loop by the integers *mod N*. The code will be invariant under a shift of m positions if the code graph connections are themselves m-shift invariant. That is, if for any information bit v_B that connects to a set of parities with indexes $i_0, i_1, \ldots, i_{q-1}$, there is another information bit v'_B such that v'_B is connected to parities $i_0 + m, i_1 + m, \ldots, i_{q-1} + m$, all indexes *mod N*. The accumulator loop is invariant under any cyclic shift *mod N* (parity to parity, channel bit to channel bit), and therefore for any shift by m, so the code itself is m-shift invariant. With this invariance, the code graph can be redrawn to have a cyclic symmetry, with all information bits and all channel bits appearing in cyclic orbits of size n. For the channel bits, take every mth bit of the accumulator loop for the channel bits to form a cycle of n; for the information bits, take those are mapped to each other by the m-shift invariance. Consequently, the parity check matrix for a QCRA code can be represented as a block matrix of $n \times n$ circulants. Effectively a shift of one position *mod n* in the circulants is equivalent to a shift of m positions *mod N* in the original code.

The parity check matrix H_{sys} for the systematic QCRA code is shown in Figure 2. Let I_s be the $n \times n$ identity matrix cyclically row left-shifted by s positions, a circulant matrix with a single one in the sth row of the first column, $0 \leq s \leq n-1$. There are $N = mn$ rows consisting of m $n \times n$ circulant blocks, and $mn + mn/q$ columns, an $m \times m(q+1)/q$ block matrix of circulants. The channel bits of the accumulator loop are on the right. That each channel bit has degree 2 is reflected by the presence of only two I_s matrices (with either $s = 0$ or $s = 1$) in each block column, giving 2 ones per column. Each row of H_{sys} represents one of the parity checks in the accumulator loop. Each parity must have degree 3, and there are two ones per row due to channel bit connections, leaving one connection to information bits. Thus each row of the block matrix has exactly one I_s matrix, for some s, in the information bit columns. Finally, each information bit has degree q, and therefore each information column of the block matrix contains q shifted identity matrices. Although here we have shown the accumulator loop equations using only I_0 and a single I_1, this is only way of representing the loop. For example, all m matrices on the diagonal can be taken to be I_0 and all the off-diagonal matrices to be I_j. So long as mj is relatively prime to n, the graph will have a single accumulator loop.

4 Recursive Quasi-Cyclic Graph Construction

The challenge of QCRA construction is to choose the shifts of various matrices I_s in H so as to define a code of relatively high quality for which an iterative decoding algorithm will perform well [4]. For the sake of latter, two heuristics are valuable: 1) making the girth g, or shortest cycle, of the graph as large as possible; and 2) making the diameter d, the maximum of the minimum path length between any pair of nodes, as small as possible. A large girth assures that the initial iterations of a sum-product decoding algorithm generate statistically independent estimates. A small diameter assures that information from every received bit will contribute to the probabilistic estimate for any

other bit rapidly, after a number of iterations proportional to the diameter. The QCRA construction technique we propose is recursive, defining the code graph for a long code in terms of the code graph of a short code with the intended rate while attempting to keep the diameter as small as possible. The general recursive construction technique, here called a quasi-cyclic or, more precisely, an n-cyclic extension, is applicable to any graph, but our interest is in bipartite code constraint graphs for QCRA codes.

Consider a bipartite graph G with a set of vertices $V = V_p \cup V_b$ consisting of bit vertices and parity vertices with a set of directed edges $E = \{e | e = (v_i, v_j), v_i \in V_p, v_j \in V_b\}$ and incidence matrix H with rows indexed by vertices from V_p and columns indexed by vertices from V_b. Let g and d be the girth and diameter of G, respectively. An *n-cyclic extension* of G, G', is defined as follows. The vertices $V' = \{[v, k] | v \in V, 0 \leq k \leq n-1\}$ are ordered 2-tuples, a vertex of G followed by an integer *mod* n. The edges $E' = \{([v_i, k], [v_j, (k - f(e) mod\ n)]) | (v_i, v_j) = e \in E, 0 \leq k \leq n-1\}$ for some function $f(\cdot)$ defined on the set of edges mapping each edge of G to the integers *mod* n. The function $f(\cdot)$ will be called the *offset function* . In brief, each vertex of G is replaced by n copies, each edge of G by a bundle of n edges, and the offset function $f(\cdot)$ determines a cyclic shift or offset of the bundle that occurs as it passes between the two expanded vertices of G. In matrix terms, the incidence matrix H' of G' is obtained by replacing each "1" in H corresponding to edge e by a matrix I_s, where $s = f(e)$. Graph G' is specified entirely by G, n, and the offset function $f(\cdot)$.

It follows immediately from the specification that each of the n copies of a vertex v has the same degree as v. For example, if G is the graph for an RA code and each parity has degree three, then the parities in G' will also have degree three. It is also easy to prove that the girth satisfies $g' \geq g$, since any closed cycle in G' passes through a series of vertices that project (by dropping the integer component) to a series of vertices forming a closed cycle in G. A cycle in G' is a cycle in G such that a running total computed by subtracting the integer $f(e)$ along an edge from a parity vertex to a bit vertex and adding $f(e)$ along an edge from a bit to a parity is zero *mod* n around the cycle. The goal in extending a graph so as to increase girth is to choose $f(\cdot)$ so that none of the cycles of length g can have a running total of zero *mod* n.

The diameter d' of G' can be much larger than d. Clearly $d' \geq d$, by again projecting: the shortest path in G' between any two vertices $v'_1 = [v_1, k_1]$ and $v'_2 = [v_2, k_2]$, where v_1 and v_2 are at distance d in G, must be at least d, since its projection is a path in G. The goal in constructing G' to limit the increase in the diameter is to try to ensure that, for an arbitrary pair of vertices of G and any desired offset $k \pmod n$, there is some path of length d' or less between them along which the cumulative offset achieves k. This motivates the investigation of QCRA constructions endowing the graph with a large automorphism group. The automorphism group facilitates the investigation of how all possible cumulative offsets can be achieved along paths between two vertices of G. The automorphism group of an n-cyclic extension has a cyclic subgroup of order n, by design, but it is possible to introduce even stronger fruitful symmetry.

5 QCRA Definition and Subgroups on $GF(p)$

For any prime p, let a be a primitive element of the Galois field $GF(p)$, so that the multiplicative order of a is $p-1$, $a^{p-1} = 1$, and any non-zero elements of the field can be written a^i, $0 \leq i \leq (p-1)$. If q divides $p-1$, $qz = (p-1)$, and a^z is an element of order q, $(a^z)^q = a^{zq} = 1$. The non-zero elements of the field can organized into an array

A with q rows and z columns such that $A(i,j) = a^{iz+j}, 0 \le i \le (q-1), 0 \le j \le (z-1)$. The entries of A will be used to define the function $f(\cdot)$ in a p-cyclic extension. A is a coset decomposition using a subgroup of order q of the multiplicative group of $GF(p)$.

Let G be a graph for a QCRA code consisting of an accumulator loop containing $p-1$ parity nodes alternating with $p-1$ channel bits. Let the parity vertexes be indexed by the integers *mod* $(p-1)$, for notational clarity written as a function $v_P(i)$, and likewise for channel bits $v_C(i)$ The jth input information bit vertex, $v_B(j)$ of degree q, is connected to parities $v_P(k), k = 0+j, z+j, 2z+j, \ldots, (q-1)z+j$, all indexes *mod* $(p-1)$. This renders the graph shift-invariant under any cyclic shift of the accumulator loop taking parities to parities and channel bits to channel bits. As a starting point, this code appears curiously weak. The girth of the graph is eight, and the minimum distance of the code is q, achieved by making two successive information bits non-zero, no matter how large p. The simple form of G comes into play, however, in showing that a quasi-cyclic extension can be a graph with good connectivity. A p-cyclic extension for $p = 13, q = 3$ and $z = 3$ is illustrated in Figure 3.

Our p-cyclic extension graph of G, G', is the constraint graph for a length $N = nm$ QCRA code, with $n = p$ and $m = p-1$. Critical to the definition, the function $f(e)$ is

$$f((v_P(iz+j), v_B(j))) = A(i,j) = a^{iz+j}, 0 \le i \le (q-1), 0 \le j \le (z-1) \tag{1}$$

which can also be written

$$f((v_P(k), v_B(k \bmod z))) = a^k, 0 \le k \le (p-1) \tag{2}$$

The offset $f(e)$ for the edge from any parity to any channel bit is zero, except for the 0th parity to $p-2$th channel bit, where $f((v_P(0), v_C(p-2))) = 1$. (We will henceforth assume all index arguments of $v_B(\cdot)$ are *mod* z, all index arguments of $v_P(\cdot)$ and $v_C(\cdot)$ are *mod* $(p-1)$.)

This one anomalous value is needed to form a single accumulator loop with $p(p-1)$ channel bits. Alternatively, any function values that have a net sum around the accumulator loop not divisible by p will suffice.

6 Graph Properties

The diameter of a QCRA graph thus defined depend on the differences in indexes that are induced by the array A. We start by considering the modified graph G'' in which the anomalous value is also set to zero, $f((v_P(0), v_C(p-2))) = 0$, and later compare the diameter of G' to that of G''. G'' has an isolated accumulator loop of length $p-1$ for each value $0 \le k \le (p-1)$ and those loops are interconnected solely by the information bits. As for G' the vertices of G'' are $V'' = \{[v,k] | v \in V, 0 \le k \le p-1\}$ are ordered 2-tuples, a vertex of G followed by an integer *mod* p. The edges of G'' from parities to information bits are the same; those from parities to channel bits are the same, excepting the anomalous edges $([v_P(0),k],[v_C(p-2),k]), 0 \le k \le p-1$ in G''.

Besides the shift invariance in the p-cycles, the automorphism group of G'' also includes a permutation of vertices that is one-transitive on the information nodes.

Proposition: The permutation π that maps $[v_B(j),k]$ to $[v_B(j+1),ak]$, $[v_P(iz+j),k]$ to $[v_P(iz+j+1),ak]$, and $[v_C(iz+j),k]$ to $[v_C(iz+j+1),ak]$, $0 \le k \le p-1$, preserves the edges of G''.

Proof: The offset function $f(e)$ has been chosen so that cyclic shift in G accompanied by multiplication of all offset values by a leaves the graph and offset values invariant.

There is only one non-trivial condition to check. Repeating π z times maps $[v_B(j), k]$ to $[v_B(j), a^z k]$, and maps $[v_P(iz + j), k]$ to $[v_P((i + 1)z + j), a^z k]$. By the definition above, $(v_P((i+1)z+j, v_B(j)))$ is an edge in G and $f((v_P((i+1)z+j), v_B(j)) = a^{(i+1)z+j} = a^z a^{iz+j}$ as required for consistency. The parity to channel bit edges are also preserved by π, and the offset function for those edges is zero per the definition of G'', so multiplying by a preserves the function values. Therefore, π is an automorphism of G''.

The p-cyclic extension ensures that, by definition, the shift σ mapping each node $[v(j), k]$ to $[v(j), k + 1]$ is an automorphism of G''. Together σ and π are one-transitive on information bit vertexes and one-transitive on parity vertexes and on channel bit vertexes. To bound the diameter of the graph, we will start by bounding the length of the shortest path between two parity nodes. Our definition of the array A permits a quick but weak upper bound to be established, and we now sketch the argument. In G'' any node $[v_P(j), k]$ can be connected to any other parity node with the same k by a path of length less than $(p - 1)$, because both nodes are part of an accumulator loop of length $2(p - 1)$. We will exhibit one path from a node $[v_P(j), k]$ to another arbitrary $[v_P(j'), k + \Delta]$. Observe that $[v_P(j), k]$ is connected via $[v_B(j \ mod \ z), k]$ to $[v_P(iz + j), k + a^j(a^{iz} - 1)]$ for $1 \leq i \leq (q - 1), 0 \leq j \leq (p - 2), 0 \leq k \leq (p - 1)$. For each of the $(q - 1)$ non-zero values of i, the equation $a^j(a^{iz} - 1) = \Delta$ has a unique solution for j, $j = log_a(\Delta) - log_a(a^{iz} - 1)$, the discrete logarithm on $GF(p)$, for any non-zero difference Δ. Thus there are $(q - 1)$ distinct paths of length two connecting nodes in the set $\{[v_P(j), k] | 0 \leq j \leq (p - 2)\}$ to nodes in the set $\{[v_P(j), k + \Delta] | 0 \leq j \leq (p - 2)\}$ for any $0 \leq \Delta \leq p - 1$. Moreover, stated non-rigorously, the discrete logarithm gives a pseudo-random distribution to j for $1 \leq i \leq (q - 1)$, which helps avoid the formation of a large number of short cycles in the graph.

The offset $f(e)$ was chosen purposefully to give the quasi-cyclic extension this dispersion property. Any $(p-1)$ parity node segment of the accumulator loop of G' is multiply connected to any other $(p - 1)$ parity node segment. In this sense, the offset function induces a graph similar to that of a difference set in cyclic coding theory [5](pp. 397-98); a rich set of differences is created between information bit tap connections, causing information to propagate rapidly in an iterative decoding process. With this alone we can give a weak upper bound on graph diameter.

Proposition: The diameter of the quasi-cyclic extension graph G' satisfies
$d \leq 4(p - 1) + 4$.

Proof: First, in G'' any parity node $[v_P(j), k]$ is at most $2(p-1)+2$ steps from any other $[v_P(j'), k']$, as follows: There is a path along the accumulator loop of length $(p-1)$ or less (counting both channel bit nodes and parity nodes) going from $[v_P(j), k]$ to a $[v_P(j_1), k]$ which is connected via a single information bit to a parity $[v_P(j_2), k']$, by the preceding discussion. (In fact, there are $(q - 1)$ choices for $v_P(j_1)$.) Then there is a path, again of length $(p - 1)$ or less, from $[v_P(j_2), k']$ to $[v_P(j'), k']$. The concatenated path has length $2(p-1)+2$ or less. In the actual quasi-cyclic extension graph G', the accumulator loop is not broken up into length $2(p - 1)$ segments; rather, it is one unbroken chain. However, there exist a path in G' from $[v_P(j), k]$ to a $[v_P(j_1), k]$ of length at most $2(p - 1)$ that does not cross one of the segment boundaries, with $[v_P(j_1), k]$ again connected via a single information bit to a parity $[v_P(j_2), k']$, and then a path of length at most $2(p - 1)$ that does not cross a segment boundary to $[v_P(j'), k']$, a total length of at most $4(p - 1) + 2$. This path is the same in both G' and G''. Finally, any channel node is distance one from a parity node, as is any information bit node, which means the diameter can be at most two greater.

This bound is very weak because it does not exploit the shortest of the $q-1$ paths between segments that are available, and it ignores all paths that pass through multiple information bit nodes. For large q, these paths become very dense rapidly and should dominate in determining both the diameter and the girth of the quasi-cyclic extension graph. Here again the automorphism group of G'' is a powerful tool for analyzing the density of multiple information bit paths, entailing a study of the differences in the p-cyclic second component of a parity node that can be achieved in two or more steps.

As mentioned previously, the girth of G' is the shortest cycle in G along which the cumulative sum of the offset function values is zero. Many paths with cumulative sum zero can be found that repeat any fixed cycle in G p times, thus arriving at zero *mod* p. There are also sum zero paths obtained by passing along the same edges of G twice, in opposite directions, but these generally involve traverses of $O(p)$ along the accumulator loop of G. The intricate analysis studies the differences between the values in any column of the array A and the differences of the differences between immediately adjacent columns, and so forth for additional steps, until two distinct paths from the same starting bit of G are found to arrive at the same vertex of G with the same cumulative sum. The group property of the columns of A seems to avoid immediate short cycles, but we have no general proof that addresses the potentially subtle number theoretic questions involved.

7 Minimum Distance Arguments

One objective in describing a QCRA code algebraically is to achieve a weight distribution for the code that is superior to those resulting from those found in RA codes with random permutations. In particular, a lower bound on the minimum distance of the code would give assurance that an error floor will not be encountered prematurely at high signal to noise ratios.

Our p-cyclic QCRA codes are amenable to transform analysis over an extension field $GF(2^x)$ containing a pth root of unity, leading to a characterization of the code space as a set of $(p-1)$ interrelated cyclic codes of length p [7]. This representation can be the source of an algebraic bound on the minimum distance of the code. For example, it can be shown algebraically that the rate 1/3 length 15 code of Figure 2 has a minimum distance of 7 and is equivalent to a [15,5,7] BCH code. To date we have not succeeded in deriving a general bound that can be readily computed for arbitrary p, and with the page limits on these Proceedings in mind, we will only make a few brief observations that give insight into the design principles.

The periodicity of the quasi-cyclic definition and the simplicity of the RA code's accumulator chain, with parity check nodes of degree three, unavoidably introduces words of weight $q(p-1)$. In particular, for any j and k, setting $[v_B(j),k]=1$ and $[v_B(j),k+1]=1$ and all other input bits zero causes the accumulator chain to turn on and emit $(p-1)$ ones starting at the first tap of $[v_B(j),k]$, turning off at the first tap of $[v_B(j),k+1]$, and similarly for all q taps of $[v_B(j),k]$, giving an output channel word with weight $q(p-1)$. For this reason the period of a QCRA should not be made too small, despite the possible circuit simplifications allowed by a shorter period. For the prescription of Section 5, the minimum distance is no greater than $q\sqrt{N}$.

In addition, there are low weight words arising from setting to one all bits in the set $\{[v_B(j),a^{j+iz}]|0\le i\le(q-1)\}$ and $[v_B(2j),0]=1$ (assuming such a bit exists), giving an output word of weight greater than $(p-1)(q-1)/2$. We conclude the discussion of minimum distance with:

Conjecture: The minimum distance of the $[(p-1)p, (p-1)p/q, D_{min}]$ QCRA code of Section 5 satisfies $(p-1)(q-1)/2 \leq D_{min} \leq (p-1)q$.

8 Simulation Results

Two QCRA codes were simulated on the Additive White Gaussian Noise (AWGN) channel by Dariush Divsalar of NASA JPL, Pasadena, and compared with RA codes of the same rate and roughly similar length. The permutation used in the comparison codes were generated at random, then screened to eliminate permutations for which the information bit taps did not pass a "spread" or "dispersion" test. The shorter QCRA is a rate 1/3 of length $N = 2638 = 36 \times 73$ with $K = 876$ information bits, a variant of the prescription of Section 5 in which only columns 0 through 35 of the A matrix were used to define information bits in a 73-cyclic extension of a length 36 code. The second was a rate 1/4 with $p = 137$ and $N = 18632 = 137 \times 136, K = 4658$. All codes were decoded with an iterative decoder of the turbo style. The results are shown in Figure 4. Compared to a somewhat longer $K = 1024$ RA code, and with 20 decoder iterations, the QCRA has higher Word Error Rate (WER) and Bit Error Rate (BER) at lower E_b/N_0 but outperforms the "random permutation" RA by both measures at higher E_b/N_0. We conjecture that this behavior can be attributed to a better minimum distance for the shorter QCRA. The longer $K = 4658$ QCRA code is about 0.1 db worse in WER than its somewhat shorter RA competitor, and virtually the same in BER. Of particular note is a point at 2.05×10^{-8} that lies beneath a lower bound on the BER of the ensemble of random codes with maximum likelihood decoding. The data point represents the decoding of 1.705×10^6 blocks, a total of 3.177×10^{10} bits, with 163 uncorrected information bit errors out of 7.942×10^9 information bits, at $E_b/N_0 = 0.95$. Eight additional iterations reduced the number of bit errors further, from 163 to 2.

9 Conclusion

The use of random permutations in turbo style codes is a commonly accepted technique. Despite their numerous drawbacks, random permutations, or rather, carefully screened and selected random permutations, create a code constraint graph that is known to have a low density of short cycles and good performance with iterative decoders. This paper presents one algebraic description for a special class of very simple turbocodes, the repeat-accumulate codes, using an apparently novel recursive graph construction technique. These algebraic RA codes are quasi-cyclic, being invariant with respect to simultaneous shifts of multiple cycles of length p. Effectively, the algebra defines the permuted connections of the information bits in an RA code. The definition was devised so as to make the code constraint graph well connected, within the limitation of the very restricted degree of the RA code graph nodes. While we have not yet established a meaningful bound on the minimum distance of the codes, simulation studies indicate that our QCRA codes can have a significantly larger minimum distance than their random permutation competitors. RA codes themselves do not achieve the energy efficiencies possible with much more complex turbocode systems, but their simplicity may outweigh the energy advantage in some applications. The import of our arguments is that algebraic methodologies can be fruitful for the design of codes for iterative decoding systems that are superior to those that use purely random graph constructions.

References

[1] S. Benedetto and G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Trans. on Information Theory*, vol. 42, no. 2, pp. 409-428, March 1996.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo codes,," *Proc. 1993 IEEE International Conference on Communications*, Geneva, Switzerland, pp. 1064-1070, May 1993.

[3] Dariush Divsalar, Hui Jin, and Robert J. McEliece, "Coding Theorems for "Turbo-like Codes," *Proceedings 36th Allerton Conf. Commun., Control, and Computing*, pp. 201-210, September 1998.

[4] F. Kschischang and B. Frey, "Iterative Decoding of Compound Codes by Probability Propagation in Graphical Models," *IEEE Jour. Selected Areas in Communications*, vol. 16, no. 2, pp. 219-230, February 1998.

[5] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland,Second Printing, 1978.

[6] W. W. Peterson and E. J. Weldon,Jr., *Error-Correcting Codes*,Second Edition, MIT Press, 1972.

[7] R. M. Tanner, "A Transform Theory for a Class of Group-Invariant Codes," *IEEE Trans. on Information Theory*, vol. 34, no. 4, pp. 752-775, July 1988.